SPACE SHUTTLE
EAST COAST ABORT MODES
FOR HIGH INCLINATION LAUNCHES

THESIS

Richard K. Neufang, Captain, USAF

AFIT/GA/ENY/97D-3

19980127 023

Approved for public release; distribution unlimited

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

AFIT/GA/ENY/97D-3

SPACE SHUTTLE

EAST COAST ABORT MODES

FOR HIGH INCLINATION LAUNCHES

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

Air Education and Training Command in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Astronautical Engineering

Richard K. Neufang, B.S.

Captain, USAF

December 1997

SPACE SHUTTLE

EAST COAST ABORT MODES

FOR HIGH INCLINATION LAUNCHES

Richard K. Neufang, B.S.
Captain, USAF

Approved:

William E. Wiesel	_24 Nov 97_
Dr. William Wiesel	date
Curtis Spenny	_24 Nov 97_
Dr. Curtis Spenny	date
(signature)	_24 Nov 97_
Maj. Jeff Turcotte	date

Acknowledgments

Table of Contents

List of Figures

List of Tables

Abstract

This study investigated the possibility of abort landing the Space Shuttle at east coast airports when launched at inclinations of 51.6 degrees or more. Computer modeling was used to characterize both the Shuttle launch out of Cape Canaveral and two methods of un-powered abort descents from various points in the launch following Solid Rocket Booster (SRB) separation.

The first method used varying values of pitch and roll held constant to control the descent. By plotting the latitude and longitude of the point in the descent when the nominal landing altitude was achieved against locations of east coast airports it was found that there are indeed east coast abort opportunities for high inclination launches out of Cape Canaveral.

The second method used a constant pitch and roll until the proper heading angle to intercept a desired target airport was reached then maneuver to 0 degrees roll and 30 degrees pitch. These trajectories were attempted throughout the launch for different airports so that windows of opportunity could be established. It was shown that these windows exist but only for limited times ranging from 8 to 34 seconds. These opportunities may be expanded if further studies investigate powered and optimal control cases.

1. Background

1.1 Introduction

This chapter will give a background of the existing abort procedures and launch profile of the Space Shuttle Flight System. Future missions of the Shuttle will then be presented as well as the problem statement and method of solution of this thesis.

1.2 *Space Shuttle Flight System Launch Profile*

The Space Shuttle Flight System is composed of the Orbiter, an external liquid propellant tank, and two Solid propellant Rocket Boosters (SRB's). The Orbiter's main engines and the SRB's provide propulsion from launch to approximately 120 seconds and an altitude of 50km. The SRB's are then released leaving the Orbiter and external tank propelled by the main engines. The external tank separates once the Orbiter is injected into the required ascent trajectory ,which is typically 480 seconds, after which the Orbital Maneuvering System (OMS) completes the insertion into the desired orbit.

1.3 *Existing Abort Procedures*

Failure of one or more of the Shuttle's systems during ascent may cause the selection of an abort mode to become necessary. The type of abort will depend on the system that failed and the time in flight. The two basic types of ascent abort modes are intact and contingency . Intact aborts are intended to safely return the Orbiter to a planned landing zone while contingency aborts generally result in a ditch operation (6:1). The four existing intact abort procedures, Abort To Orbit, Abort Once Around, Return To Launch Site, and TransAtlantic Landing can return the Orbiter safely to the ground in as fast as 35 to 90 minutes from launch depending on the mode selected. Figure 1-1. demonstrates the Shuttle launch profile as well as the abort modes.
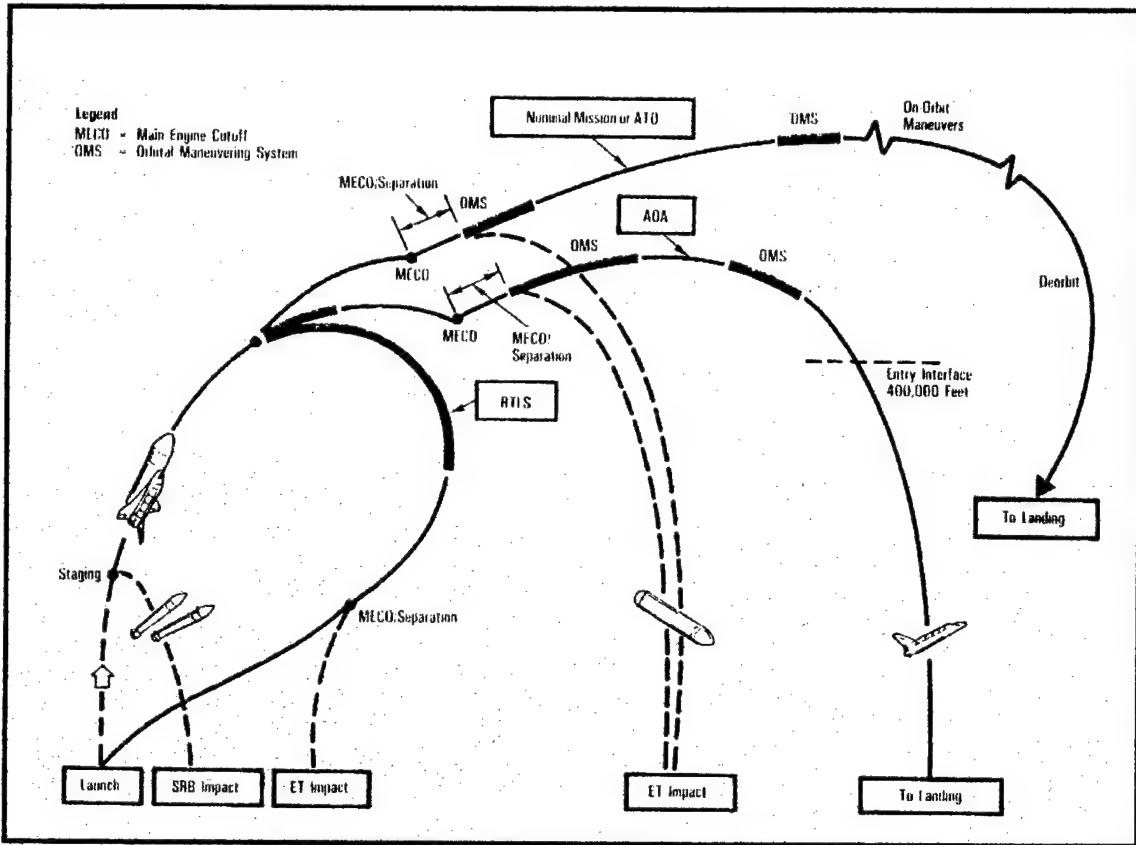
Figure 1-1. Shuttle Launch Profile and Abort Modes (6:1)

1.3.1 Abort to Orbit. An Abort To Orbit or ATO is used if the Shuttle still has sufficient

propulsion to place it into a lower orbit then planned. The OMS engines will then place it into a circular

orbit where it can deorbit as it would under normal circumstances and land at the planned site. This mode

is only chosen if there are sufficient working systems and time to land is not a factor (6:1).

1.3.2 Abort Once Around. Abort Once Around (AOA) is used when vehicle performance is

insufficient to achieve a feasible orbit or the OMS doesn't have enough fuel to perform the orbit and deorbit

maneuvering necessary for an ATO. Furthermore the AOA is used in cases where it is necessary to land as

quick as possible such as a cabin leak or cooling loss. The maneuver is done by firing the OMS once after

main engine shut off so that a second OMS thrust will send the vehicle out of orbit to land at an AOA

landing site such as White Sands, NM, Edwards AFB ,or Kennedy Space Center. This mode results in the

Orbiter circling the Earth once and landing in 90 minutes (6:1).

1.3.3 Return to Launch Site. The RTLS abort mode is selected when there is a system error during the first 4 minutes and 20 seconds of the ascent at which time there is not enough main propulsive system propellant to return to the launch site. This abort selection is also preferable if the Orbiter and crew need to be on the ground quickly. The RTLS mode can put the Orbiter on the ground in 25 minutes from launch (6:1).

The procedure for a RTLS consists of three phases; a powered phase, a separation phase, and a glide phase. The powered phase begins with the crew selection of the abort at 2 minutes 20 seconds which is just after the separation of the solid rocket boosters. The shuttle continues down range to dissipate excess main engine propellant so that it has just enough fuel to turn around, fly back towards the launch site, and achieve the proper main engine cutoff conditions so the vehicle can glide to the runway after external tank separation. As it continues downrange the Orbiter/external tank makes a pitch around maneuver so the thrust of the main engines are used to null the downrange velocity. All excess fuel like the OMS propellant are dumped at this time to improve the Orbiter weight and center of gravity. At 20 seconds prior to main engine cutoff the separation phase begins with a powered pitch down maneuver that brings the shuttle to the required external tank separation attitude and pitch rate. A reaction control system is then employed to ensure that the Orbiter does not recontact the external tank and that the Orbiter has achieved the necessary pitch attitude to begin the glide phase. Once the external tank has been released the glide phase begins which is handled like a normal entry.
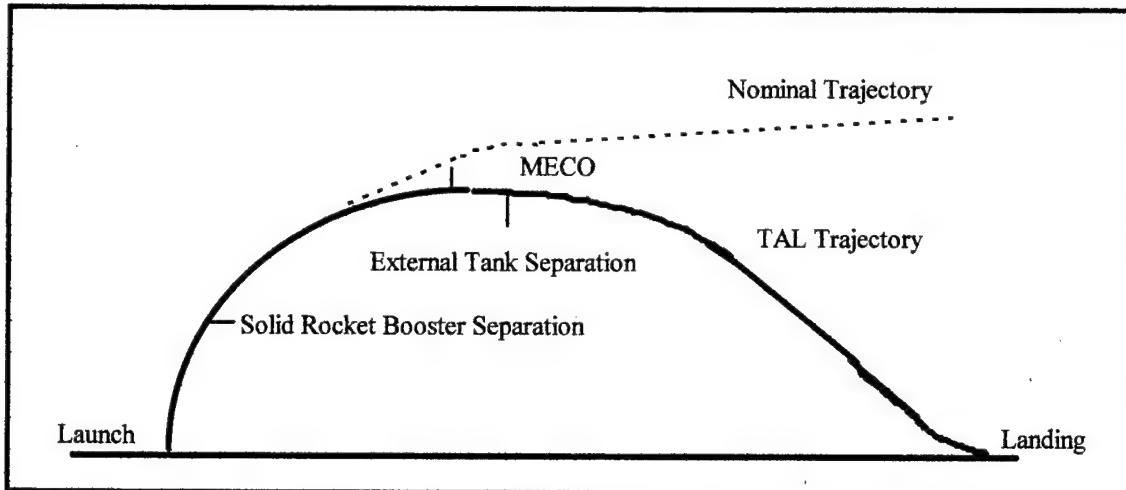
Figure 1-2. Transatlantic Abort Landing Options (6:1)

1.3.4 Transatlantic Landing. The Transatlantic Abort Landing (TAL) was developed for low inclination launches to handle emergency situations that might occur after the last RTLS opportunity and before an AOA could be attempted. This mode selection could either mean that there is insufficient power to accomplish an AOA or that the emergency calls for the Orbiter to land immediately.

In a TAL abort the vehicle would continue out of Cape Canaveral on a low inclination trajectory that would place the shuttles downrange ground track in close proximity to the West Coast of Africa or the South West of Europe. As shown in Figure 1-2 the Shuttle would continue on a ballistic trajectory across the Atlantic to land at a predetermined site that lies near the nominal ascent ground track. The site would have to have sufficient runway length , weather conditions, and approval from the State Department (6:1).

The TAL abort mode sends commands to steer the vehicle toward the plane of the landing site and roll heads up before main engine cutoff. As in an RTLS, OMS propellant is dumped to decrease weight and place the center of gravity in the proper position for control. The landing is then handled like a nominal entry.

1.3.5 Future Shuttle Missions. The Space Shuttle is scheduled to launch 50 missions through August of 2003. As shown in the pie chart in Figure 1-3., 82 percent of the launches will be flown at a 51.6 degree inclination which makes the TAL abort mode invalid because the ground track does not intercept any transatlantic landing sites (6:1). This in turn leaves between 4 minutes 20 seconds until just prior to

1-4

main engine cutoff where the Orbiter has only a contingency abort option rather than an intact abort option. The high inclination launches do, however, come significantly close to the East Coast of North America, namely the United States of America.

Percent Distribution of Launch Inclinations for Future Space Shuttle Missions

Legend:
- 39 deg
- 57 deg
- 28.45 deg
- 51.6 deg
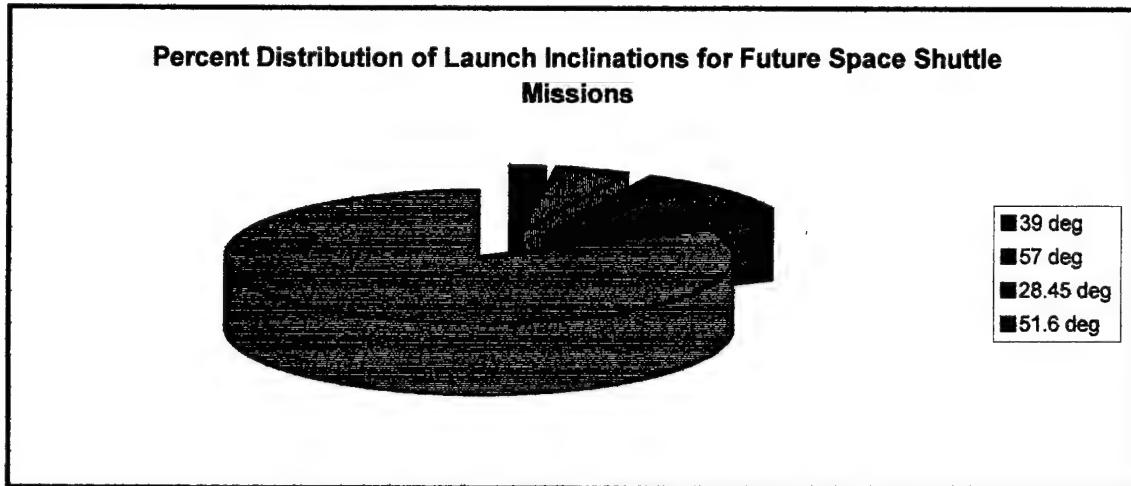
Figure 1-3. Inclination Angles for Future Shuttle Missions

1.4 Problem Statement and Method of Solution

Since the majority of the future launches out of Cape Canaveral will take the shuttle close to the coast of the United States is it possible for the shuttle to make an abort landing at an East coast airport? This thesis will explore the possibilities of safely landing the Orbiter at an East Coast airport by first creating a computer model of a shuttle launch from 51.6 degrees and 57 degrees then designing another computer model using Hypersonic Orbiter reentry data collected from NASA to mimic the dynamics of the shuttle making an abort landing. The Launch model will give the Abort model initial state vectors from 4 minutes 20 seconds until just prior to main engine cutoff. Several control methods will be used to land the Orbiter. The first method will use constant pitch and roll until the Orbiter reaches a preset altitude. The second will hold pitch and roll constant but change pitch to 30 degrees and roll to 0 degrees when the desired heading angle is achieved to intercept a target airport. Both methods will use the same abort model with minor changes in output and the afore mentioned control commands. The first method will map out

possible areas of intercept over the map of the East Coast as well as final state vectors giving longitude, latitude, speed, and altitude. The location with respect to airports of sufficient runway lengths and low enough speeds with respect to nominal re-entries will prove the feasibility of this idea as well as give an idea as to which airports the second method will use as target landing sites. The second method will map out all abort descents throughout the launch trajectory to the target airports selected from the first method. This will give clear times throughout the launch when certain airports become abort landing sites.

2. Methodology

2.1 Introduction

This chapter will explain, in detail, how the problem of examining possible abort landings at east-coast airports from launches of 51.6 and 57 degree inclinations out of Cape Canaveral is approached and resolved using computer modeling in FORTRAN. The first section explains how the launch of the Shuttle was computer modeled as a gravity-turn trajectory. This program provides state vectors at every point throughout the launch, which are used as initial points in the abort landing program. The second section discusses the abort landing computer model that uses hypersonic equations, as well as NASA data from Shuttle re-entries to take the orbiter without external tank and bring it to a nominal landing altitude using two methods of controlling the descent using pitch and roll without thrust. For both methods pitch is measured as positive nose up and roll is measured positive roll left. The first method uses constant pitch and roll angles for the entire descent while the second method only holds pitch and roll constant until a certain heading angle is reached then roll becomes 0 degrees and pitch becomes 30 degrees. Output data for both methods are mapped against airport locations which have sufficient runway lengths to accommodate the Space Shuttle. These two control methods will give a clear answer to the possibility of an east coast abort mode.

2.2 Space Shuttle Launch Model

2.2.1 Assumptions. When attempting to model reality, the first step is to minimize the number of variables so that the problem becomes workable. However, eliminating too many variables will trivialize the problem and the model will be nothing close to reality. For the launch program a flat non-rotating Earth was assumed. This sounds like a huge departure from reality, but when looking at the problem from the point of view of what is needed, i.e. the state vector, the only states which involve a round Earth are longitude and latitude. These states can be derived from the downrange distance using spherical

trigonometry and the acceleration terms in the equations of motion are modified to include the apparent "centrifugal force." So, in effect, a spherical problem is made flat.

The second assumption is that the velocity, thrust, and drag vector lie along the same line. This simply means the Shuttle always travels in the direction its nose is pointing. Furthermore, the thrust is assumed to be constant and perfectly efficient. This eliminates both the possibilities of throttling the engines and any thrust loss due to change in altitude. The coefficient of drag is assumed to be 1.0 for all components of the space shuttle system. Of course, each component will have different values of drag, due to the differences of frontal surface area.

Two more assumptions are made at the beginning of the launch since the initial state of the program starts when the Shuttle is already 100 meters off the ground. Time and mass start at 0 and M0 respectively, meaning that it took no time to fly the first 100 meters and no mass was lost getting there. Since this is a multi-stage launch, it is also assumed that solid rocket booster and external tank separation happen at exactly 120 seconds and 480 seconds, respectively.

The final assumption is that the atmosphere behaves exactly like the Regan and Anandarskarian Atmosphere Model. Figure 2-1 describes the density profile of this model which shows the density to be substantially less above 50 km altitude (9: Appendix A)
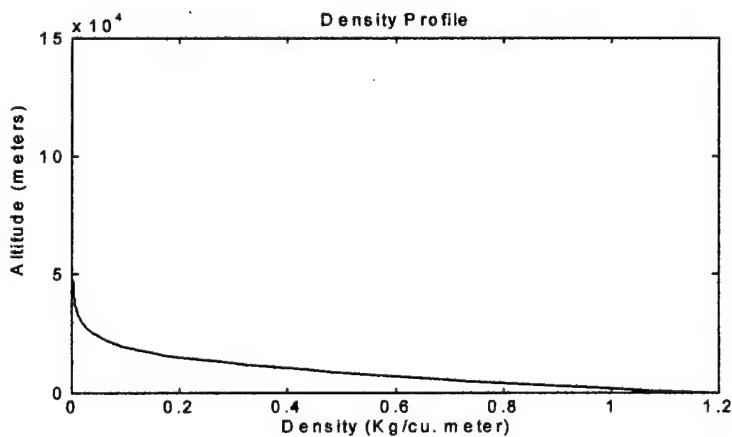
Figure 2-1. Density Profile for Atmosphere Model

2.2.2 *Statistics Used for Launch* . Table 2-1 lists Shuttle mass and performance data, as well as constants necessary for the computer program.

Table 2-1. Shuttle Statistics (7:490; 6:1)

Shuttle Mass	
Orbiter + Payload	108,864 Kg
2 Solid Rocket Boosters(SRBs)	164,000 Kg
External Tank	32,000 Kg
Fuel	1,736,336 Kg
Total Launch Mass (M0)	2,041,200 Kg
Shuttle Performance	
Thrust at Launch	30,300kN
Thrust at Second Stage	6,300kN (at 95% 6,000kN)
Mass Flow Rate at Launch	9,906 Kg/s
Mass Flow Rate at Second Stage	1,496 Kg/s
Constants	
Radius of the Earth (R)	6,378.135 Km
Gravitational Parameter of the Earth (μ)	3.98601E14 m^3/s^2
Sea Level Pressure (P0)	101,325 N/m^2
Front Surface Area of the Orbiter	29.235 m^2 (Estimated)
Front Surface Area of External Tank	50.265 m^2
Front Surface Area of Both SRBs	21.621 m^2
Coefficient of Drag (C_D)	1.0 (Assumed)
Longitude of Cape Canaveral	-80.60 degrees
Latitude of Cape Canaveral	28.46 degrees
Initial Launch Conditions	
Down Range Distance (X)	0 m
Altitude (H)	100 m
Velocity (V)	32.1 m/s
Flight Path Angle (γ)	1.56209405 radians
Mass (M)	2,041,200 Kg

2.2.3 Gravity-Turn Trajectory. The dynamics of the Space Shuttle launch model were chosen to match the equations governing a gravity turn trajectory. This type of launch aligns the thrust and velocity vectors, as mentioned in the assumptions, as well as the drag vector. This is done primarily for launch vehicles that cannot sustain large aerodynamic loads in the transverse direction, but can carry large loads in the axial direction (13:207).
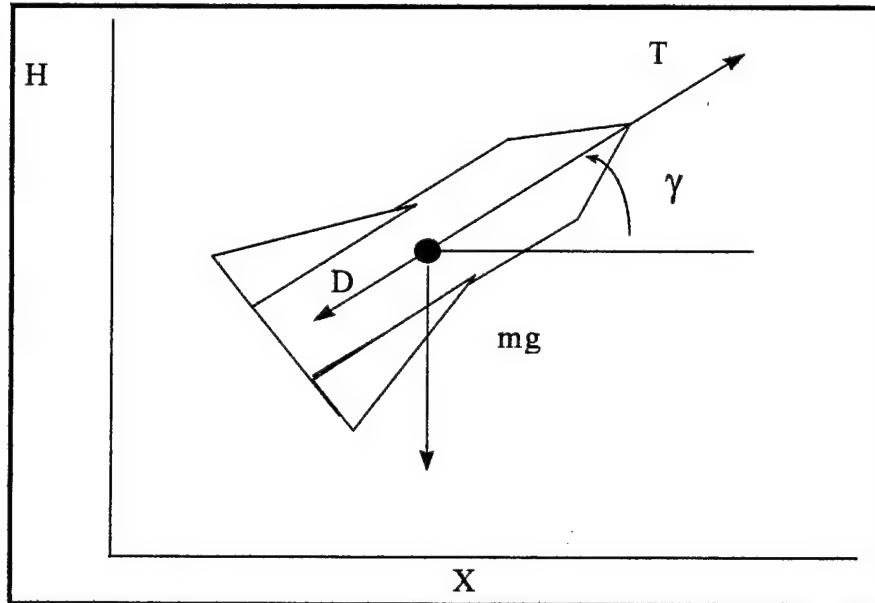
Figure 2-2. Free Body Diagram of Launch Vehicle (13:208)

Figure 2-2 shows the free body diagram of the launch vehicle with all forces acting on it. The equations of motion for this type of launch trajectory are as follows:

$$\frac{dX}{dt} = V \cos \gamma \qquad (2.1)$$

$$\frac{dH}{dt} = V \sin \gamma \qquad (2.2)$$

$$m\frac{dV}{dt} = T - D - (mg - \frac{m\dot{X}^2}{R+H})\sin\gamma \qquad (2.3)$$

$$mV\frac{d\gamma}{dt} = -(mg - \frac{m\dot{X}^2}{R+H})\cos\gamma \qquad (2.4)$$

where

$X = $ *Downrange Distance* (m)

$H = $ *Altitude* (m)

$V = $ *Speed* (m/s)

$\gamma = $ *Flight Path Angle* (radians)

$M = $ *Mass* (kg)

$T = $ *Thrust* (Newton's)

$D = $ *Drag* (Newton's)

$R = $ *Radius of the Earth* (m)

$t = $ *Time* (s)

The only noticeable irregularity in these equations are the bracketed terms in equations 2.3 and 2.4. These terms include apparent "centrifugal force" so that flight over a spherical Earth can be resolved to one over a flat Earth (13: 208). These equations must be modified further to accurately model an actual launch with statements pertaining to the change in mass, thrust, and drag with time. Both mass flow and thrust were obtained from Space Shuttle data and the drag, which is discussed later in this chapter, was estimated.

The first step in numerically integrating these equations is to find the initial conditions. The initial launch can't start with a flight path angle of 90 degrees. Otherwise, by equation 2.4, γ would never change. This is, of course, not desired so a slight "nudge" is necessary to cause γ to change and by the negative sign in equation 2.4 it will decrease, causing the vehicle to fall over. If this is done on the pad, the booster

would be lying on the ground in a matter of seconds. To prevent this from happening, the pitch over or "nudge" is performed at 100 meters altitude. This is chosen because it is best to do the initial pitch over at low speed, since the vehicle will briefly have an angle of attack during the maneuver, and high speeds would inflict greater aerodynamic loads.

After the initial altitude is chosen, the velocity can be determined by integrating the launch in a straight line from 0 to 100 meters. Knowing the thrust, mass, and drag on the space shuttle, the initial velocity was calculated to be 32.1 m/s. As mentioned in the assumptions, mass was assumed not to change over this period. Now that all the initial conditions, except for γ have been found, the only thing left to do is set final conditions and iterate different initial values of γ until those final conditions are met.

Final conditions were established at each stage of the Shuttle's launch profile by data from NASA. At the first stage, altitude was given to be 50 km and velocity had to be 1.4 km/s (10: 4). At main engine cut-off, γ had to be 0, while altitude and velocity had to be 145 km and 7.82 km/s, respectively. The velocity was calculated from the altitude of 145 km using the equation for burn-out velocity of a circular orbit given by

$$V_{burnout} = (\frac{\mu}{R + H_{MECO}})^{1/2} \tag{2.5}$$

where

$\mu = Gravitational\ Parameter\ for\ the\ Earth\ (m^3/s^2)$

$R = Radius\ of\ the\ Earth\ (m)$

$H_{meco} = Altitude\ at\ Main\ Engine\ Cut-Off\ (m)$

These conditions were used as a reference to model the launch so that optimally the launch should look like Figure 2-3(13: 210).
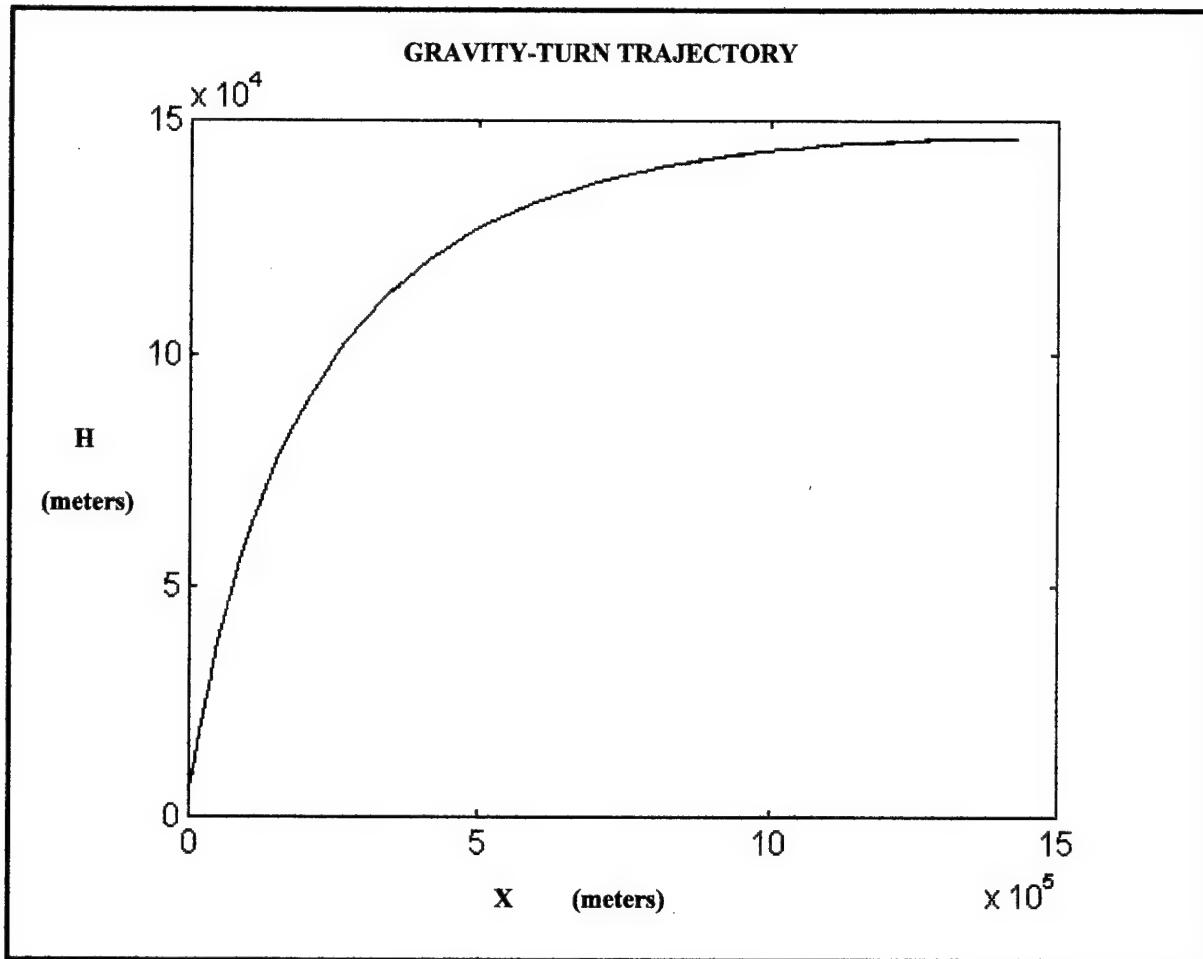
Figure 2-3. Gravity-Turn Trajectory

However, when the equations of motion were numerically integrated an angle for γ could not be found to match both staging conditions. This is most likely do to the fact that in reality the Shuttle throttles it's engines and since this model uses constant thrust a slight modification must be made. A control input for γ is used at various points to achieve both conditions so the actual computer model's launch looks like Figure 2-4.
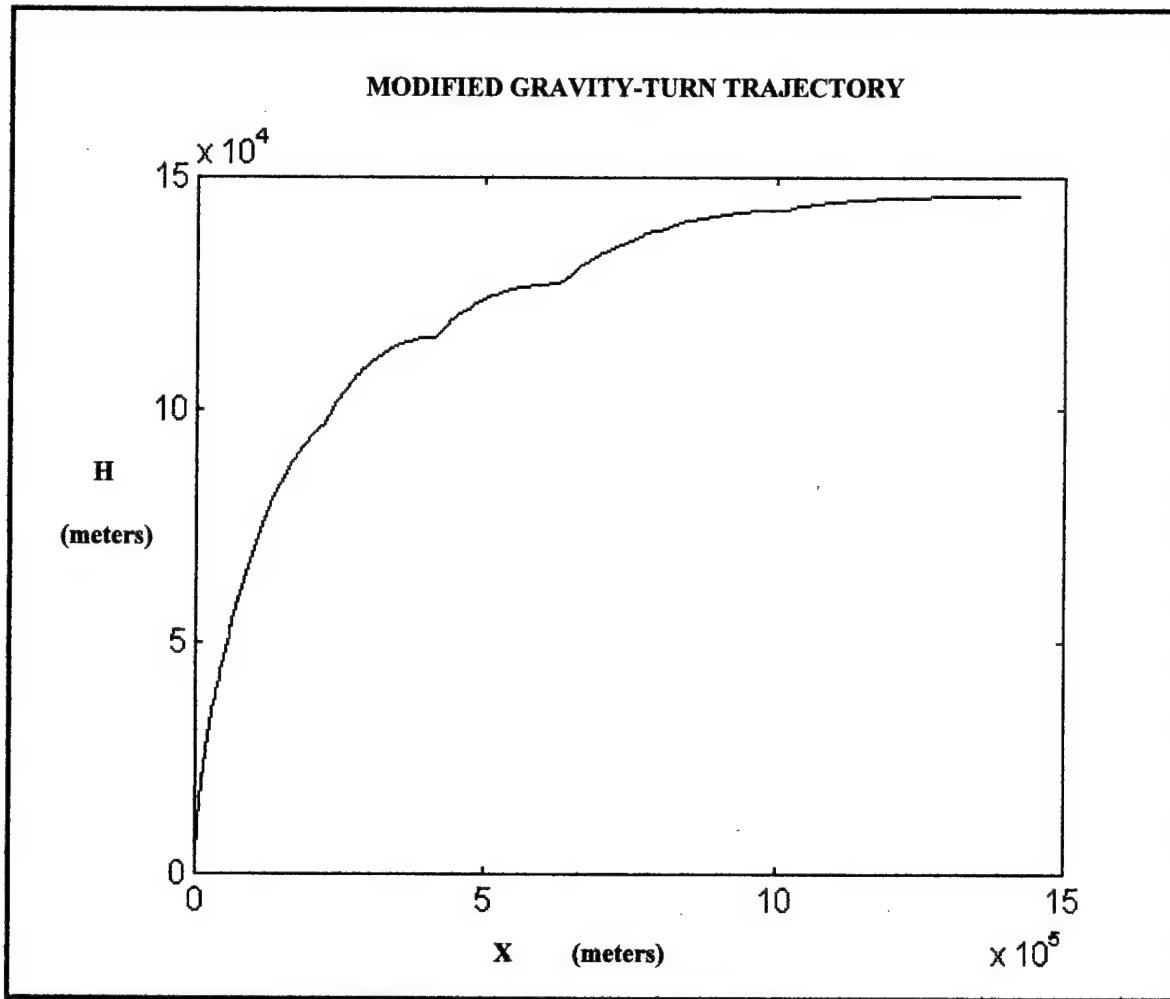
Figure 2-4. Modified Gravity-Turn Trajectory

This model doesn't exactly match the optimum trajectory but provides a close enough approximation to give reasonable state vectors necessary for the abort program. The value for the initial γ was found to be 1.56209405 radians with an increase of .11 radians at 210 seconds, .12 radians at 280 seconds, .10 radians at 340 seconds, and .02 radians at 420 seconds.

    2.2.4 Drag. Before numerical integration is possible, it is first necessary to define the drag term in equation 2.3. The drag law states that

$$D = \frac{1}{2} C_D A \rho V^2 \qquad (2.6)$$

where

C_D = *Coefficient of drag* (assumed to be 1.0)

A = *Frontal surface area* (m^2)

V = *Speed* (m/s)

ρ_0 = *Atmosphere density (kg/m^3)*

The frontal surface areas of the SRBs and external tank were calculated from the equation

$$A = \frac{d^2 \pi}{4} \qquad (2.7)$$

where

d = *Diameter* (m)

The frontal area of the Orbiter could only be estimated by adding the area of both the fuselage and the wings. The frontal area of the wings were estimated by multiplying the wingspan, 24m, by an assumed value for the wing thickness, 0.4 m.

The atmosphere density was found using a FORTRAN subroutine called Atm, which, given altitude and ground level pressure, outputs density at altitude. This subroutine uses the atmosphere model found in the appendix of Regan and Ananderskarian, as mentioned in section 2.2.1.

The other parameter, coefficient of drag, is also mentioned in 2.2.1. Although C_D is known to be a function of mach number, Reynolds number, and aerodynamic configuration, it is assumed as constant. The values of C_D may range from 0.12 to 2.0, depending on the aforementioned conditions, however 1.0 was selected as a mid-range value as well as for simplicity (1:55).

2.2.5 Numerical Integration. The equations of motion and initial conditions for the gravity-turn are now known. What is needed now is to first put equations 2.1-2.4 into the form

$$\dot{X} = f(x,t) \qquad (2.8)$$

which is done simply by dividing equation 2.3 by the mass and equation 2.4 by both mass and velocity. The next step is to ensure that the state vector elements have close to the same order of magnitude. SI units

were chosen for both their ease in conversions and proper orders of magnitude. The last step is then to find a numerical integrator which is best suited for the particular problem.

Numerical integrators fall into several classes: extrapolators, predictors, and predictor-correctors. Extrapulators and predictors step their way into the future using data from the current instant and the immediate past (12:119). Predictor-correctors add a corrector algorithm which runs a higher order polynomial through the previous data points to obtain a better value of the new state vector (12:123). From these choices, a fourth order predictor-corrector subroutine in FORTRAN called Haming was selected (Appendix A). Although Haming is over fifteen years old, it still performs better than ODE on the VAX or CYBER by a substantial margin (12:120). Since this integrator is fourth order, the only drawback is that the initial state provides only one point and an initialization method is needed to get the other three before they can be called into operation.

In addition, like most integrators, Haming prefers smooth functions and since this is a multi-stage launch where mass and thrust change instantaneously at various stages, Haming must be re-initialized at each stage. To accomplish this, it was necessary to write three different subroutines called Rhs1, Rhs2, Rhs3, which all include equations 2.1-2.4 but have different thrust and mass values corresponding to stage. These three subroutines are then integrated using integrators Haming1, Haming2, Haming3 and added together. This provides a smooth integration over a function with several jumps.

2.2.6 Spherical Trigonometry. The original state vector, [X, H, V, γ, M], having gone into the program and been numerically integrated is output as that state at any time along its flight. However, the desired output is to have location measured in longitude and latitude on the globe rather than downrange distance, X. We also want to know the heading angle at every instant after launch. This step, in effect, turns the 2D launch into three dimensions. Latitude, longitude, and heading angle can be computed using the original outputs and few added initial states.

The first step to modifying these outputs is to calculate the initial heading angle, ψ_0, given the inclination of the launch and the longitude and latitude of the launch site. Figure 2-5 displays a right triangle made from three great circles (8:20).
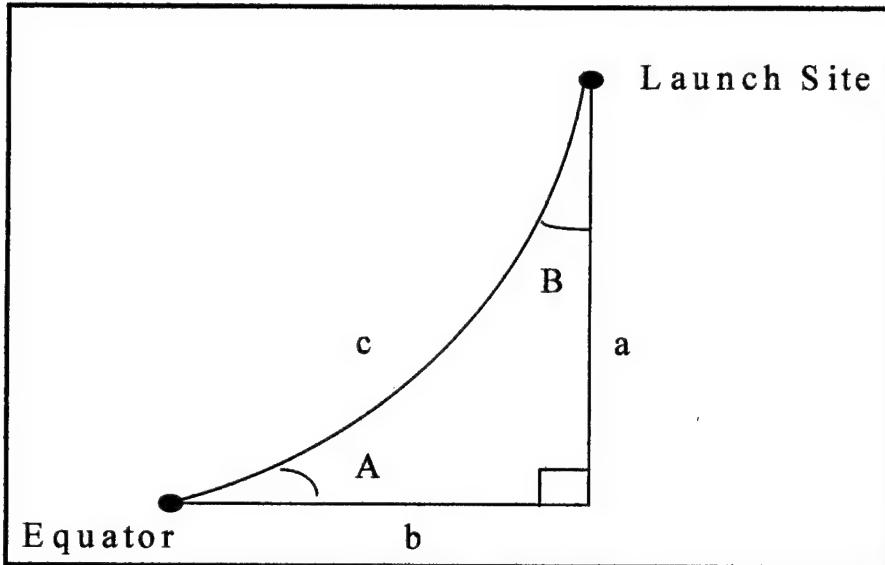
Figure 2-5. Spherical Triangle from Equator

The two vertices that are not 90 degrees represent the equator and the launch site, and the inscribed angles,

A and B, represent the launch inclination and azimuth angle respectively. The two sides, a and b, are the

latitude of the launch site and the longitudinal distance from the equatorial crossing. From Napier's Rule, it

is found that

$$\cos A = \cos a \sin B \qquad (2.7a)$$

or rearrange

$$B = \sin^{-1}\frac{\cos A}{\cos a} \qquad (2.7b)$$

The initial heading angle can then be found by subtracting the azimuth from 90 degrees.

The next step is to find the longitude and latitude throughout the flight, given the conditions

already stated and the downrange distance.
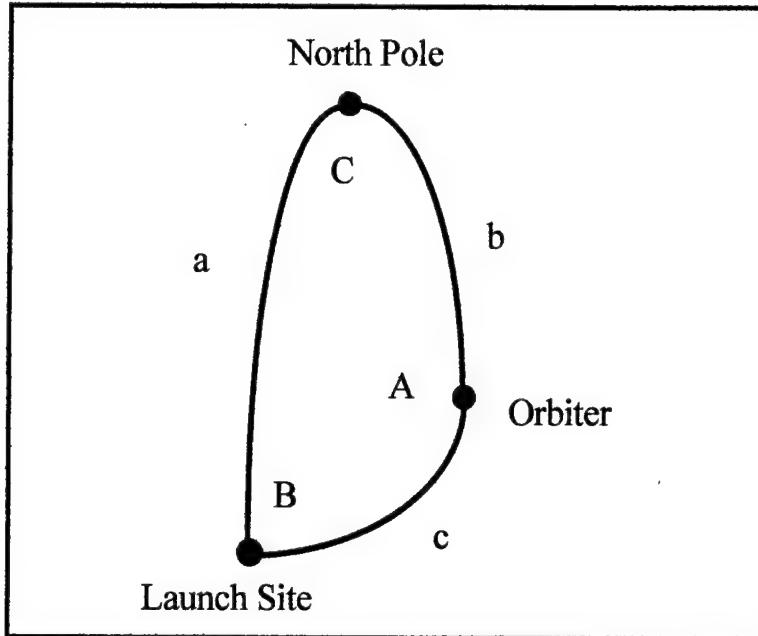
Figure 2-6. Spherical Triangle from Launch Site

Figure 2-6 shows yet another spherical triangle with the three vertices as the launch site, Orbiter, and North Pole. Side c is the downrange distance converted to radians by dividing by the radius of the earth making it a great circle. The other known values are a , which is the difference of 90 degrees and the latitude of the launch site, and B which is the launch azimuth found in equation 2.7b. The following equations are used to solve for the unknowns

$$b = \cos^{-1}\left[\cos a \cos c + \sin a \sin c \cos B\right] \qquad (2.8)$$

$$C = \cos^{-1}\left[\frac{\cos c - \cos b \cos a}{\sin b \sin a}\right] \qquad (2.9)$$

where

b = *Difference of 90 Degrees and the Latitude of the Orbiter*

C= *Difference in the Longitude of the Launch Site and the Longitude of the Orbiter*

Then all that remains is to solve both b and C for the latitude and longitude of the Orbiter.

$$Lat_{orbiter} = 90 - b \qquad (2.10)$$

2-12

$$Long_{orbiter} = Long_{site} - C \qquad (2.11)$$

where

Lat_{site} = Latitude of launch site

$Long_{site}$ = Longitude of launch site

The heading angle can then be evaluated at every point in the flight by subtracting equation 2.7b from 90

degrees with 'a' as the latitude of the launch vehicle, rather than that of the launch site.

2.2.7 *Flow Chart for Gravturn Program.* Figure 2-7 shows how each subroutine mentioned in the

previous sections of this chapter interface with each other to provide the necessary output.
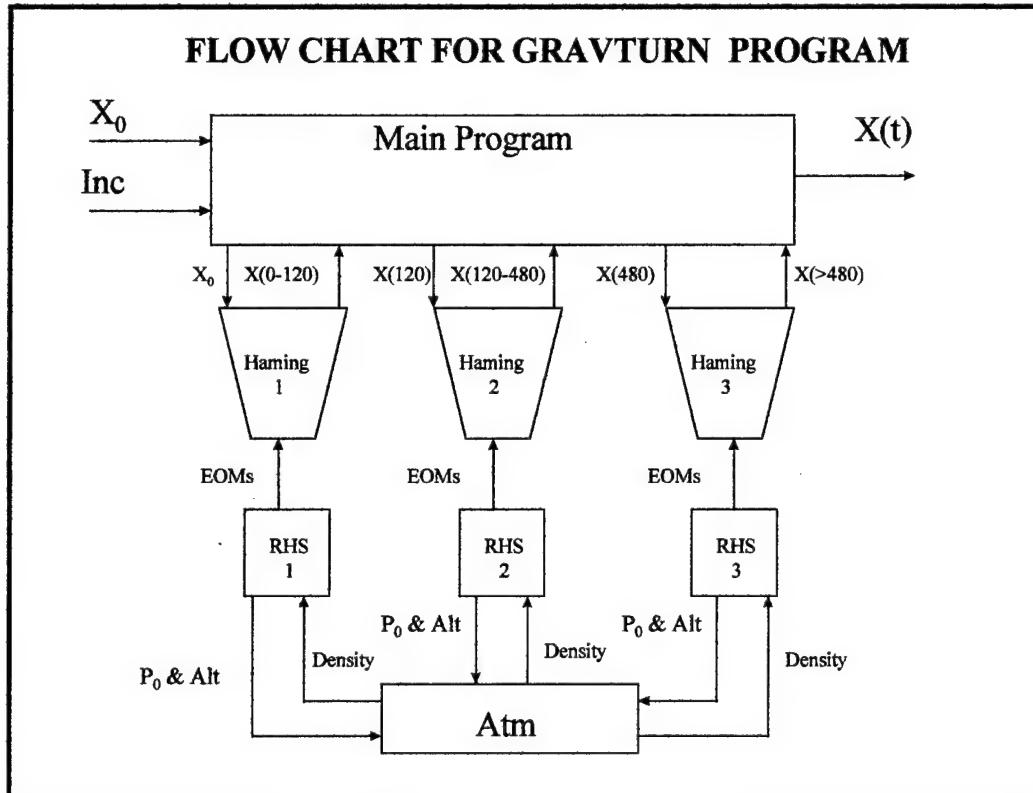
Figure 2-7. Flow Chart of Gravturn Program

The inputs are the angle of inclination, along with the initial state vector: downrange distance, altitude,

velocity, flight path angle, and mass. The output is the new state vector at any time during the flight:

altitude, longitude, latitude, velocity, flight path angle, and heading angle. The main program and all subroutines can be found in Appendix A of this thesis.

2.3 Abort Landing Model

Many different circumstances may cause the Shuttle to choose an abort mode. These modes vary depending on which systems have failed and if time to land is critical. This thesis cannot explore every possible combination of working and malfunctioning systems, but it will attempt to resolve the feasibility of aborting without any working propulsion system given the flight control system still operates. The abort program will take the Shuttle at any point in it's launch after SRB separation, drop the external tank, and glide in using only pitch and roll to control the un-powered descent. This gives a clear idea as to which times throughout the launch, if any, it will be possible to land at an east coast airport.

2.3.1 *Assumptions*. For this computer model, the thrust is assumed to be zero. Since the external tank is dropped, the mass will be only that of the Orbiter plus payload. The propulsion system is non-operative and cannot expend fuel, meaning the mass is constant. The control system on board is operative, which leads to the largest assumption, an aerodynamic model of the Orbiter.

The Rarefied - Flow Shuttle Aerodynamics Flight Model will be assumed to provide C_L and C_D values for varying altitudes, speeds, and vehicle attitudes. This model is taken from NASA data collected on twelve Orbiter re-entries and will be discussed later in this chapter. Although the field of rarefied-flow is highly theoretical, it is based on actual data and should serve as the best assumption available for modeling winged vehicles through atmosphere at hypersonic speeds.

Further assumptions are made to use equations based on flight over a spherical planet. One is that the planet is rotating at such a rate that the coriolis acceleration term, $2\omega V$, is retained but the higher order terms such as $\omega^2 r$ are neglected. The second is that the Orbiter is treated as a point mass. This only means the forces acting on the Orbiter are of primary interest whereas the moments can be looked at in a more in depth study if east coast aborts are indeed considered feasible.

2.3.2 *Statistics Used For Shuttle Abort Model*. Table 2-2 shows the values used to represent the performance and specifications of the Shuttle for the Abort Program.

Table 2-2. Data Used for Abort Program (3:550-555; 2:429; 13:316)

Shuttle Data	
Orbiter & Payload Mass (M)	108,864 Kg
Vehicle Reference Area (S)	249.9 m^2
Mean Aerodynamic Chord (MAC)	12.058 m
Constants	
Radius of the Earth (R)	6378.135 Km
Gravitational Parameter of the Earth (μ)	3.98601x10^{14} m^3/s^2
Rotation Rate of the Earth (ω)	7.292115856x10^{-5} radians/s
Sea Level Pressure (P$_0$)	101325 N/m^2
Final Conditions for Nominal Landing	
Altitude	25,000 m
Velocity	800 m/s

2.3.3 *Flight Over A Spherical Planet*. This section will derive and explain the equations of motion (EOMs) that were used in the abort landing program. As mentioned in the assumptions, these equations govern the motion of a point mass with zero thrust flying inside an atmosphere of a spherical planet. They describe the time rate of change of the radius, longitude, latitude, velocity, flight path angle, and heading angle.

Figure 2-8 shows a planet fixed system XYZ and another system xyz which is rotating with respect to it.
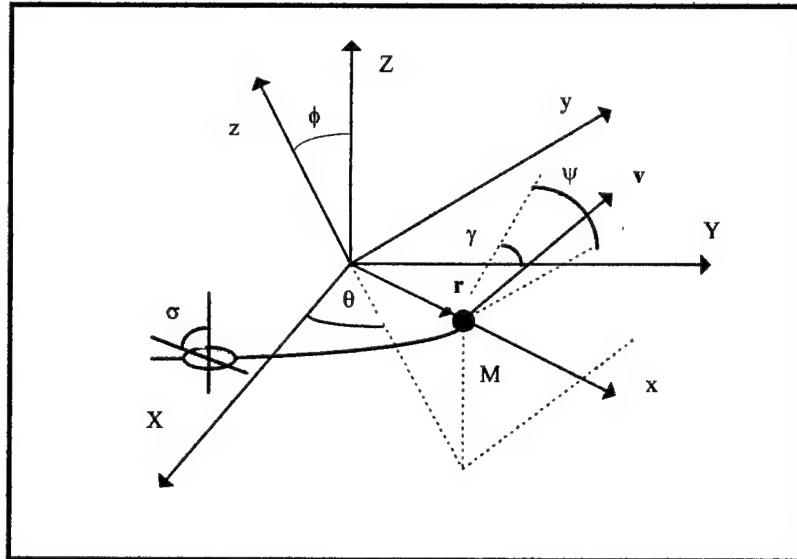
Figure 2-8. Coordinate Systems (11:22)

The planet fixed system is rotating with the angular velocity of the earth, ω, assumed constant and directed along the Z - Axis (11:21). The vectors **r** and **v** represent the position and velocity of the point mass M. The angles θ and ϕ represent the longitude (measured from the X Axis, in the equatorial plain, and positively eastward) and latitude (measured from the equatorial plane, along a meridian, and positively northward) respectively. As in the gravity-turn trajectory γ and ψ are the flight path angle and heading angle respectively. The flight path angle is measured between the local horizontal and the velocity vector while the heading angle is between the local parallel of latitude and the projection of **v** on the horizontal plane. The angle σ or roll angle is measured from the direction of lift from the Orbiter and the plane made by the position and velocity vector (11: 23-24).

If **i** , **j** , and **k** are unit vectors along the xyz reference frame, the position, velocity, and angular velocity of the earth can be written in component notation as

$$\vec{r} = r\hat{i} \tag{2.12}$$

$$\vec{v} = v\sin\gamma\ \hat{i} + v\cos\gamma\cos\psi\ \hat{j} + v\cos\gamma\sin\psi\ \hat{k} \tag{2.13}$$

$$\vec{\omega} = \omega\ \sin\phi\ \hat{i} + \omega\cos\phi\ \hat{k} \tag{2.14}$$

where

r = *Magnitude of the Radius* (m)

v = *Magnitude of the Velocity* (m/s)

ω = *Angular Velocity of the Earth* (radians/s)

γ = *Flight Path Angle* (radians)

ψ= *Heading Angle* (radians)

ϕ= *Latitude* (radians)

It follows that acceleration can be found using the transport theorem

$$\frac{d\vec{v}}{dt}_{inertial} = \frac{d\vec{v}}{dt}_{local} + 2\omega\ x\ \vec{v} + \omega\ x(\omega\ x\ \vec{r}) \tag{2.15}$$

where

$\dfrac{d\vec{v}}{dt}_{local}$ = *Time Rate of Change of Velocity wrt the Local Reference Frame*

Substituting Eq 2.15 into Newton's second law results in

$$M\frac{d\vec{v}}{dt}_{local} = \vec{F} - 2M\vec{\omega}\ x\ \vec{v} - M\omega\ x\ (\omega\ x\ \vec{r}) \tag{2.16}$$

where

M = *mass* (kg)

F = *Total Force Vector* (Newton's)

The total force is represented in the free body diagram of figure 2-9. The force of gravity, mg, is in the negative x direction while the force of drag is in the negative direction of the velocity vector. All that remains to have a complete knowledge of all the forces involved is to determine the direction of lift in the xyz coordinate frame.
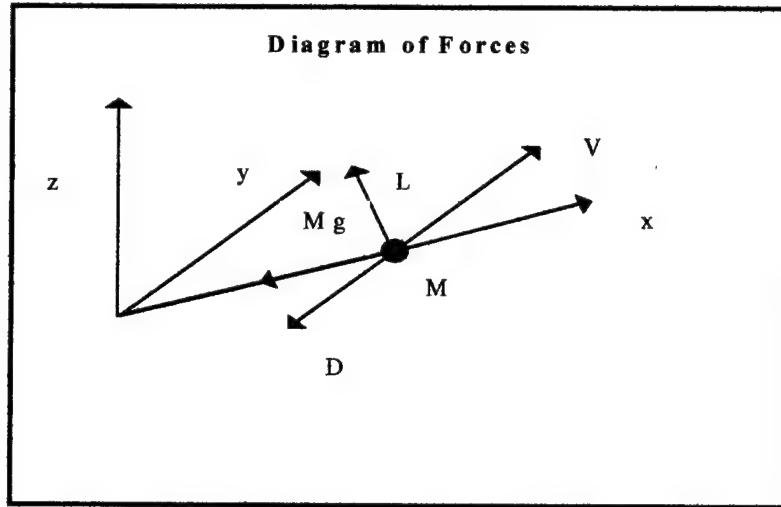
Figure 2-9. Force Diagram

Let $x_1y_1z_1$ be an intermediate coordinate system rotated an angle σ from a body fixed frame as shown in figure 2-10.
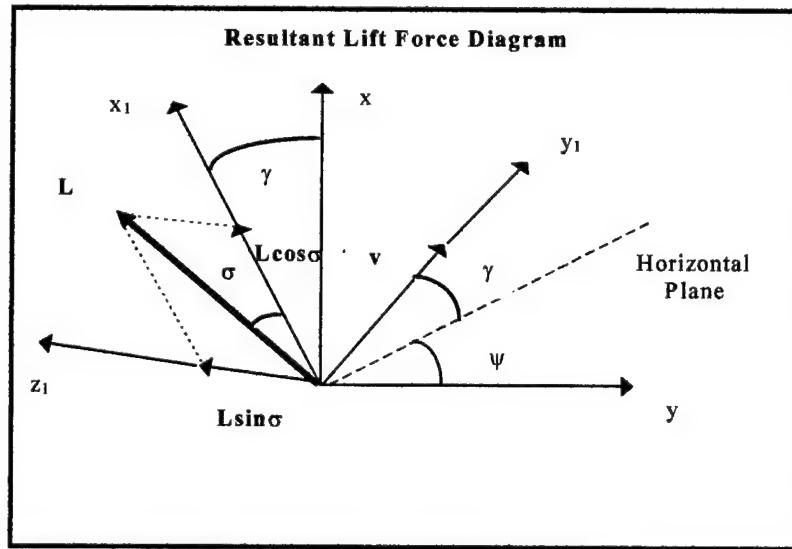
Figure 2-10. Lift Force Diagram

The system $x_1y_1z_1$ is deduced by a rotation ψ in the horizontal plane then a rotation γ in the vertical plane of the xyz system (11:24). The transformation matrix equation for this sequence of rotations is

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \cos\gamma & \sin\gamma & 0 \\ -\sin\gamma\ \cos\psi & \cos\gamma\ \cos\psi & -\sin\psi \\ -\sin\gamma\ \sin\psi & \cos\gamma\ \sin\psi & \cos\psi \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \qquad (2.17)$$

which resolves the direction of lift in the xyz system to

$$\vec{L} = L\cos\sigma\ \cos\gamma\ \hat{i} - (L\cos\sigma\ \sin\gamma\ \cos\psi + L\sin\sigma\ \sin\psi)\vec{j}$$
$$- (L\cos\sigma\ \sin\gamma\ \sin\psi + L\sin\sigma\ \cos\psi)\vec{k} \qquad (2.18)$$

The total force, position, velocity, and angular velocity of the earth are now known in terms of the unit vectors, **i, j, k** in the xyz frame. In order to take the derivatives of the vectors with respect to the XYZ frame, the derivatives of the unit vectors **i, j, k** must be determined.

In much the same fashion as the earth fixed referenced frame XYZ was evaluated in an inertial frame by means of the transport theorem via the rotation rate of the earth, ω, the xyz frame must be evaluated via its rotation rate, Ω, relative to the inertialy fixed planet. The system xyz is acquired from the XYZ system by a rotation θ about the positive z axis then by a rotation ϕ about the negative y axis (11:24). This yields the angular velocity

$$\vec{\Omega} = (\sin\phi\frac{d\theta}{dt})\hat{i} - (\frac{d\phi}{dt})\hat{j} + (\cos\phi\frac{d\theta}{dt})\hat{k} \qquad (2.19)$$

The time derivatives can then be obtained using Poisson formulas (11:20)

$$\frac{di}{dt} = \vec{\Omega} \times \hat{i} = (\cos\phi \frac{d\theta}{dt})\hat{j} + (\frac{d\phi}{dt})\hat{k}$$

$$\frac{dj}{dt} = \vec{\Omega} \times \hat{j} = -(\cos\phi \frac{d\theta}{dt})\hat{i} + (\sin\phi \frac{d\theta}{dt})\hat{k} \qquad (2.20)$$

$$\frac{dk}{dt} = \vec{\Omega} \times \hat{k} = -(\frac{d\phi}{dt})\hat{i} - (\sin\phi \frac{d\theta}{dt})\hat{j}$$

The next step is to take the time derivatives of equations 2.12 and 2.13, which for position yield

$$\frac{d\vec{r}}{dt}_{inertial} = (\frac{dr}{dt})\hat{i} + (r\cos\phi \frac{d\theta}{dt})\hat{j} + (r\frac{d\phi}{dt})\hat{k} \qquad (2.21)$$

which can be broken up by component unit vectors and set equal to equation 2.13 yielding

$$\frac{dr}{dt} = v\sin\gamma \qquad (2.22)$$

$$\frac{d\theta}{dt} = \frac{v\cos\gamma\cos\psi}{r\cos\phi} \qquad (2.23)$$

$$\frac{d\phi}{dt} = \frac{v\cos\gamma\sin\psi}{r} \qquad (2.24)$$

Equations 2.22, 2.23, and 2.24 represent the first three equations of motion used in the abort program. The second three equations of motion are obtained from taking the derivative of equation 2.13 and substituting the first three EOMs for $\frac{dr}{dt}$, $\frac{d\theta}{dt}$, and $\frac{d\phi}{dt}$. This result along with the values for total force and ω are

placed into Eq 2.16, the force equation. This equation is then rearranged to solve for $\dfrac{dv}{dt}$, $\dfrac{d\gamma}{dt}$, and $\dfrac{d\psi}{dt}$.

After neglecting terms which have $\omega^2 r$, the final three EOMs are

$$\frac{dv}{dt} = \frac{-D}{M} - g\sin\gamma \qquad\qquad (2.25)$$

$$\frac{d\gamma}{dt} = \frac{L\cos\sigma}{Mv} - \frac{g\cos\gamma}{v} + \frac{v\cos\gamma}{r} + 2\omega\cos\phi\cos\psi \qquad\qquad (2.26)$$

$$\frac{d\psi}{dt} = \frac{L\sin\sigma}{Mv\cos\gamma} - \frac{v\cos\gamma\cos\psi\tan\phi}{r} + 2\omega(\tan\gamma\cos\phi\sin\psi - \sin\phi) \qquad\qquad (2.27)$$

where

$D = drag$ (Newton's)

$L = lift$ (Newton's)

$M = mass$ (kg)

$g = gravity$ (m/s^2)

These six equations were placed in a subroutine called Dynam where they can be numerically integrated by the Haming subroutine used in Gravturn once the values for Drag and Lift are determined.

2.3.4 *Shuttle Aerodynamics Flight Model.* Equations 2.22-2.27 describe the dynamics for the abort model and are known but can not be numerically integrated. Equations 2.25, 2.26, and 2.27 have lift and drag terms whose equations are

$$L = \frac{1}{2}C_L\rho\ v^2 S \qquad\qquad (2.28)$$

$$D = \frac{1}{2}C_D\rho \; v^2 S \tag{2.29}$$

where

C_L = coefficient of lift

C_D = coefficient of drag

ρ = density (kg/m^3)

v = velocity (m/s)

S = reference area (m^2)

Density is found using the same earth atmosphere model of figure 2-2. However, the coefficients of lift and drag can't be assumed constant; therefore a model must be found to accurately estimate the lift and drag coefficients the Orbiter will experience at various altitudes and speeds. The obvious choice is the Shuttle rarefied - flow aerodynamics flight model that was derived from a High Resolution Accelerometer Package, HiRAP, on twelve Shuttle re-entries spanning a period of ten years (6:550).

Rarefied - flow is the transition region between free molecular flow and hypersonic continuum, which is roughly 160-60 km in altitude. These regions are determined based on the Knudsen number, Kn, or ratio of the mean free path to the mean aerodynamic chord, MAC (6: 552). The regions are clearly separated in Table 2-3.

Table 2-3. Atmospheric Regions Based on Knudsen Number (3:550)

Kn	Region
$<10^{-3}$	Hypersonic Continuum
$10^{-3}<Kn<10$	Rarefied-Flow
>10	Free Molecular Flow

Since the majority of the time the Orbiter will be in the rarefied - flow region it will be necessary to use the empirical equations which govern it. The equations are split into three segments: Hypersonic continuum, Free molecular flow, and a bridging formula. All three segments calculate C_N and C_A, which are aerodynamic coefficients in the normal and axial direction, as functions of pitch angle, α. The reason for using normal and axial rather than lift and drag is primarily due to the placement of the HiRAP on board the Orbiter. This only adds one further step which will convert C_N and C_A to C_L and C_D.

The hypersonic continuum equations for the normal and axial coefficients as a function of pitch are

$$C_{Nc} = -9.25704 \times 10^{-5} \alpha^2 + 5.23808 \times 10^{-2} \alpha - 0.839782 \tag{2.30}$$

$$C_{Ac} = 5.86689 \times 10^{-7} \alpha^3 - 6.72027 \times 10^{-5} \alpha^2 + 3.32044 \times 10^{-3} \alpha - 0.0086314 \tag{2.31}$$

these equations are curve fit to match Shuttle data for an envelope of $35 < \alpha < 45$ degrees (3:552).

The free molecular flow equations for C_N and C_A as a function of α are

$$C_{NF} = -7.16528 \times 10^{-6} \alpha^3 + 9.66197 \times 10^{-4} \alpha^2 + 9.18422 \times 10^{-3} \alpha + 1.58739 \times 10^{-3} \tag{2.32}$$

$$C_{AF} = -1.17117 \times 10^{-5} \alpha^3 + 5.92205 \times 10^{-4} \alpha^2 + 0.0164864 \alpha + 0.751105 \tag{2.33}$$

These equations were curve fit to match pitch angles between 0 and 60 degrees (3:553). All that remains to compute C_N and C_A for the Shuttle is the bridging formula to combine hypersonic continuum with free molecular flow.

The bridging formula depends on the value of Kn as shown in the following equations

$$\overline{C}_N = \exp\left[-0.29981(1.3849 - \log_{10}Kn)^{1.7128}\right]$$
$$if \quad \log_{10}Kn < 1.3849 \tag{2.34}$$
$$otherwise \quad \overline{C}_N = 1.0$$

$$\overline{C}_A = \exp\left[-0.2262(1.2042 - \log_{10}Kn)^{1.8410}\right]$$
$$if \quad \log_{10}Kn < 1.2042 \tag{2.35}$$
$$otherwise \quad \overline{C}_A = 1.0$$

Now the results from equations 2.30 - 2.35 are used to calculate the final normal and axial aerodynamic coefficients of the Orbiter

$$C_N = C_{Nc} + (C_{NF} - C_{Nc})\overline{C}_N \tag{2.36}$$

$$C_A = C_{Ac} + (C_{AF} - C_{Ac})\overline{C}_A \tag{2.37}$$

The normal and axial coefficients can now be converted to C_L and C_D. Figure 2-11 shows a vector representation of the alignment of these coefficients on a vehicle with an angle of attack.
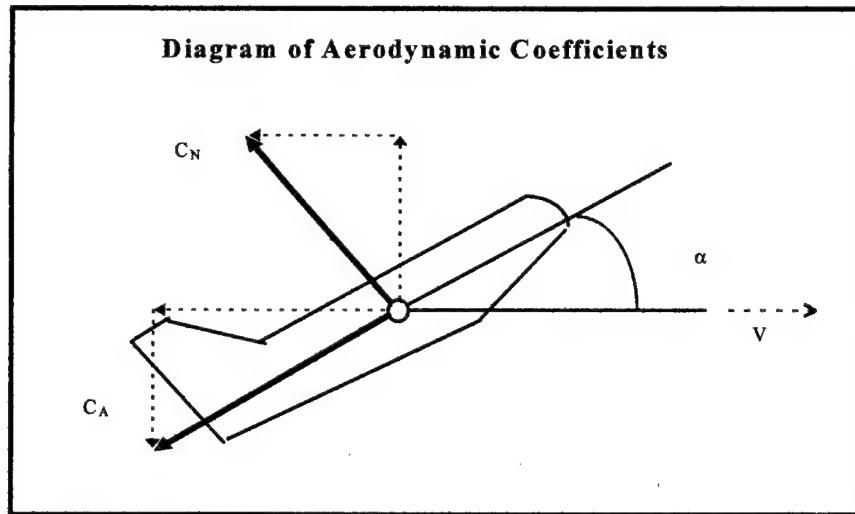
Figure 2-11. Vector Components of Aerodynamic Coefficients

The dashed components make up C_L and C_D

$$C_L = -C_A \sin\alpha + C_N \cos\alpha \qquad (2.38)$$

$$C_D = C_A \cos\alpha + C_N \sin\alpha \qquad (2.39)$$

The equations from 2.30 to 2.39 were placed in a subroutine called Aero which returns C_L and C_D given pitch and Kn. This in turn provides the EOMs placed in the subroutine Dynam with lift and drag so that they may now be numerically integrated much the same way the gravity turn trajectory equations were.

2.3.5 *Flow Chart.* Figure 2-12 shows how the abort program interacts with it's subroutines.
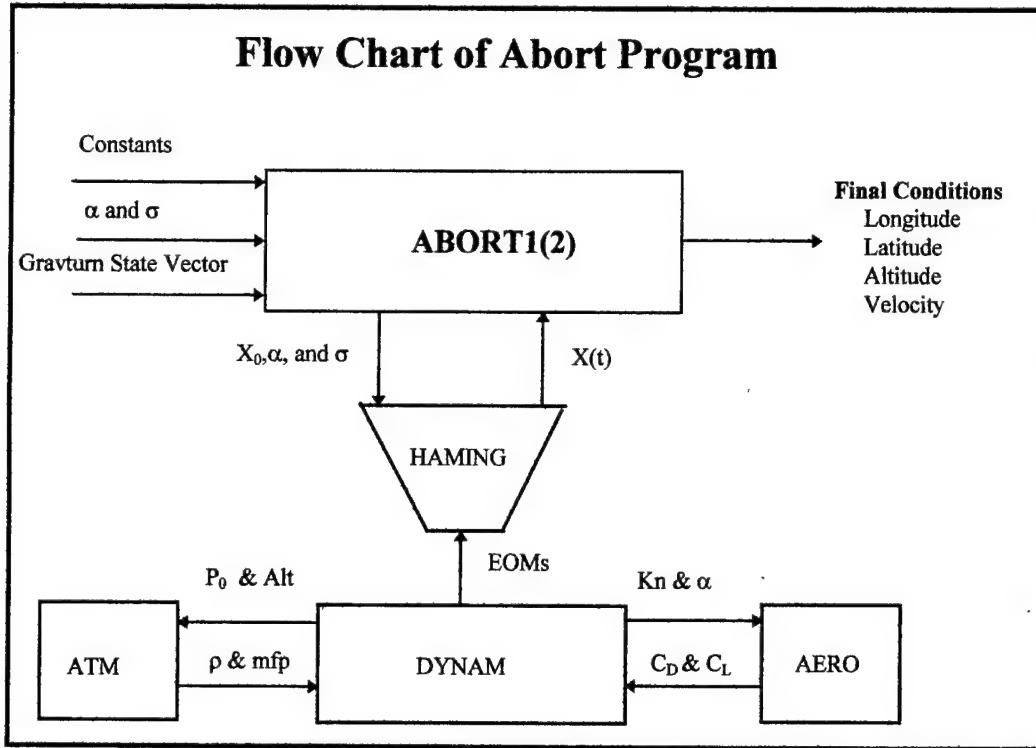
Figure 2-12. Flow Chart Of Abort Program

The inputs to the program are the state vector produced by the Gravturn Program, the constants necessary to the reentry equations, and varying values of pitch and roll. The program then calls Haming which in turn calls Dynam. Dynam contains the six EOMs that describe flight over a spherical planet which need density and the coefficients of lift and drag that it receives from Atm and Aero respectively. The output of the program provides a final longitude, latitude, altitude, and velocity which are mapped against existing airports to determine if east coast aborts are possible.

3. Data Analysis and Results for Constant Control Descent

3.1 Introduction

This chapter presents the data from the Abort1 program from six points of the 51.6 and 57 degree inclination launches generated by the Gravturn program. From each of the six points, constant pitch and roll angles are cycled through every combination between 30 and 90 degrees at 5 degree increments for pitch and 0 and 90 degrees for roll and numerically integrated up to the nominal landing altitude of the orbiter. At this point the longitude and latitude is plotted against the map of the Eastern United States creating landing zones and the average velocity of each landing zone is compared to the nominal Shuttle landing speed. Airport locations are then added to compare intercept points for possible abort landings.

Results drawn from the data determine whether the constant control descent proves the feasibility of an east coast abort or if further control methods are necessary.

3.2 Gravity-Turn Trajectory Launch

Numerically integrating the equations described in 2.2.3 and plotting the latitude and longitude at every 2 seconds throughout the launch gives a picture of the launch trajectory. When plotted with a computer generated map of the Eastern United States it demonstrates just how close the ascent brings the orbiter to the east coast. Figures 3-1 and 3-2 show the launch of a 51.6 and 57 degree inclination launch out of Cape Canaveral from 0 to 480 seconds.
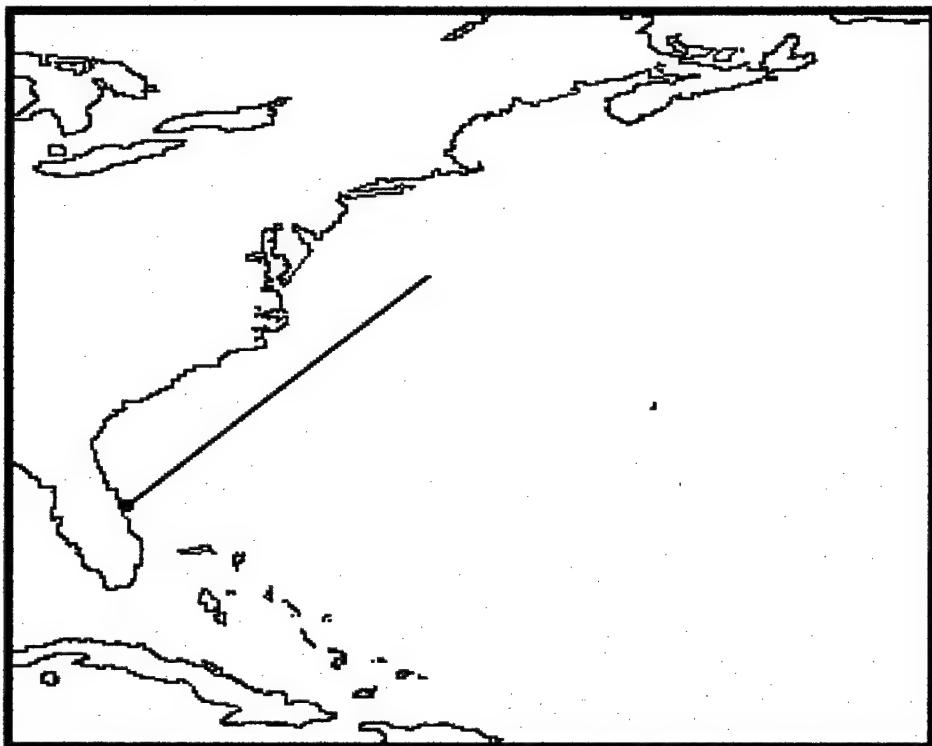
Figure 3-1. 51.6 Degree Inclination Launch from Cape Canaveral
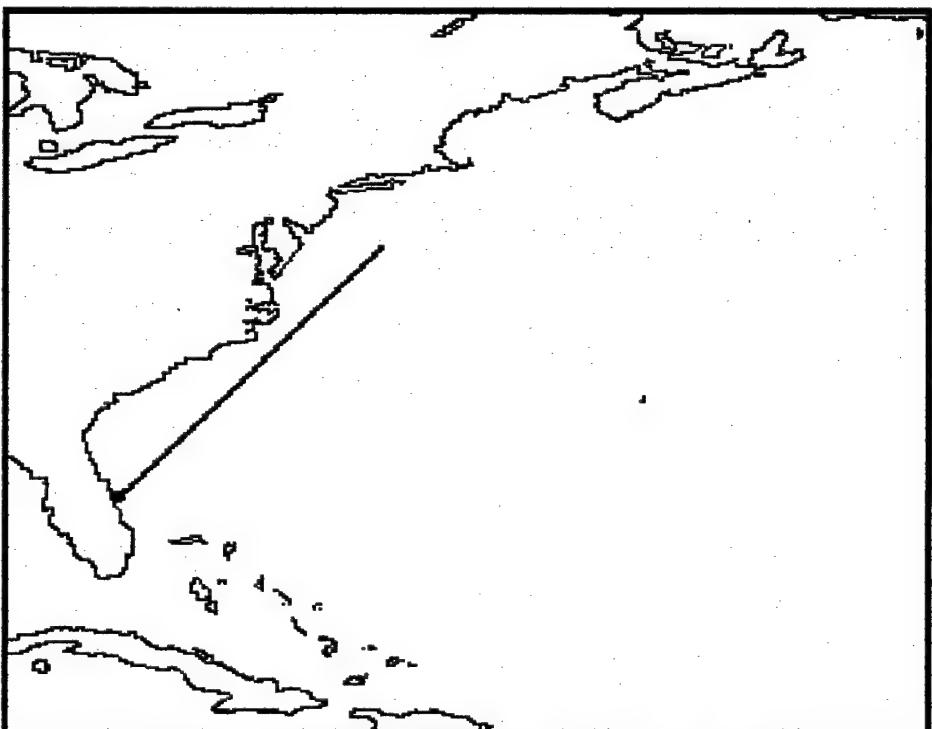
Figure 3-2. 57 Degree Inclination Launch from Cape Canaveral

3-2

After the launch was plotted six points throughout the launch were chosen whose state vector would be used as the initial state for the Abort program. Since nothing can be done until SRB separation the first abort point was selected just afterwards at 120 seconds. The remaining five points were picked at 60 second intervals of 180, 240, 300, 360, and 420 seconds. Figures 3-3 and 3-4 show these six abort points mapped to the 51.6 and 57 degree inclination launches shown in Figures 3-1 and 3-2.
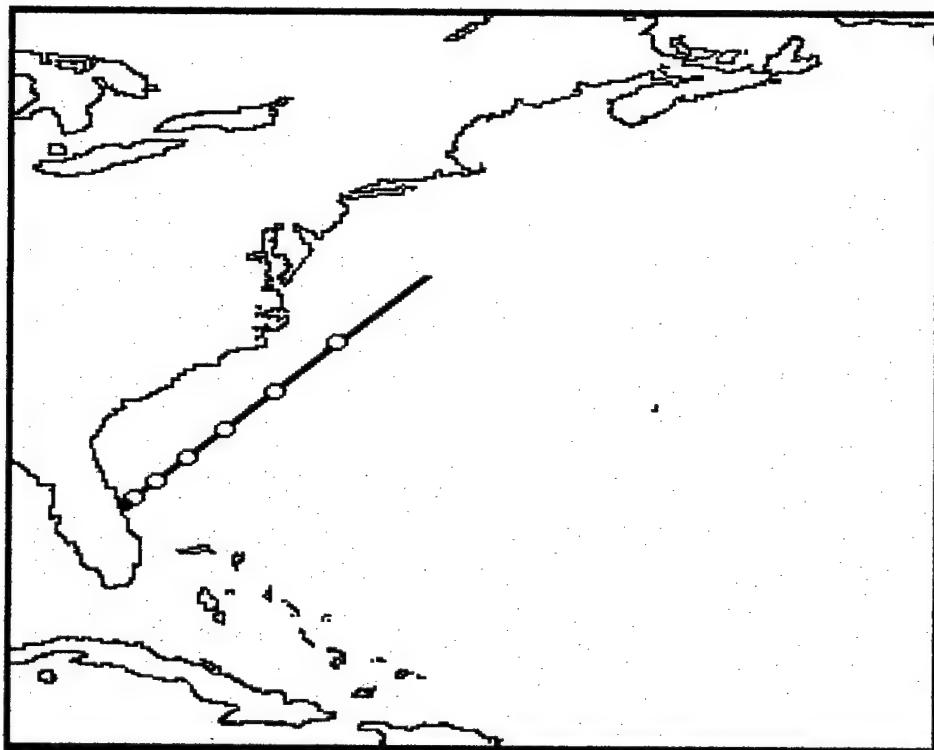
Figure 3-3. Six Abort Points on 51.6 Degree Inclination Launch
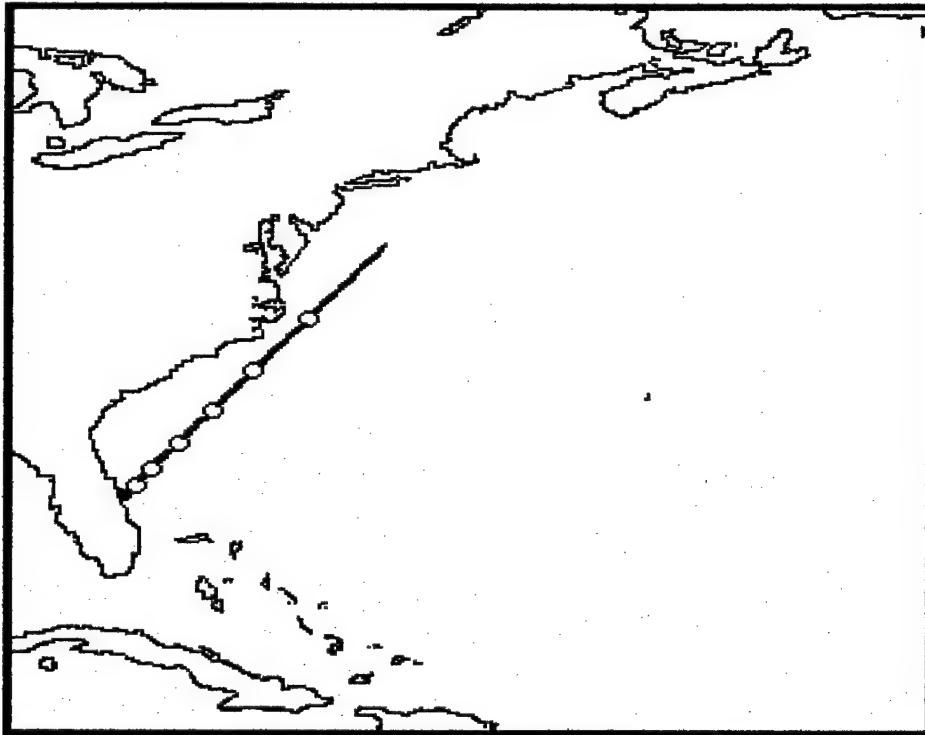
Figure 3-4. Six Abort Points on 57 Degree Inclination Launch

3.3 Abort Landing

The Abort1 program uses the six abort points as initial states to numerically integrate equations 2.22 through 2.27. Pitch and roll angles are held constant throughout the descent, but cycled through all possible combinations ranging from 30-90 degrees of pitch and 0-90 degrees of roll for each abort point. The ranges in pitch exceed the angle parameters established by the equations in the aerodynamics model discussed in 2.3.4 but provide all possibilities within the Shuttle's flight envelope. This range increase simply assumes that the aerodynamics equations apply for increased angles of 45 degrees in the hypersonic continuum and 30 degrees in free molecular flow. The equations are integrated using all these angular combinations until an altitude of 25,000 meters is reached, which is the nominal landing altitude of the Orbiter. The final latitude and longitude of each combination is then plotted on the computer generated map of the Eastern United States to form six different zones. These landing zones are shown in figures 3-5 and 3-6 for both a 51.6 and 57 degree inclination launch.
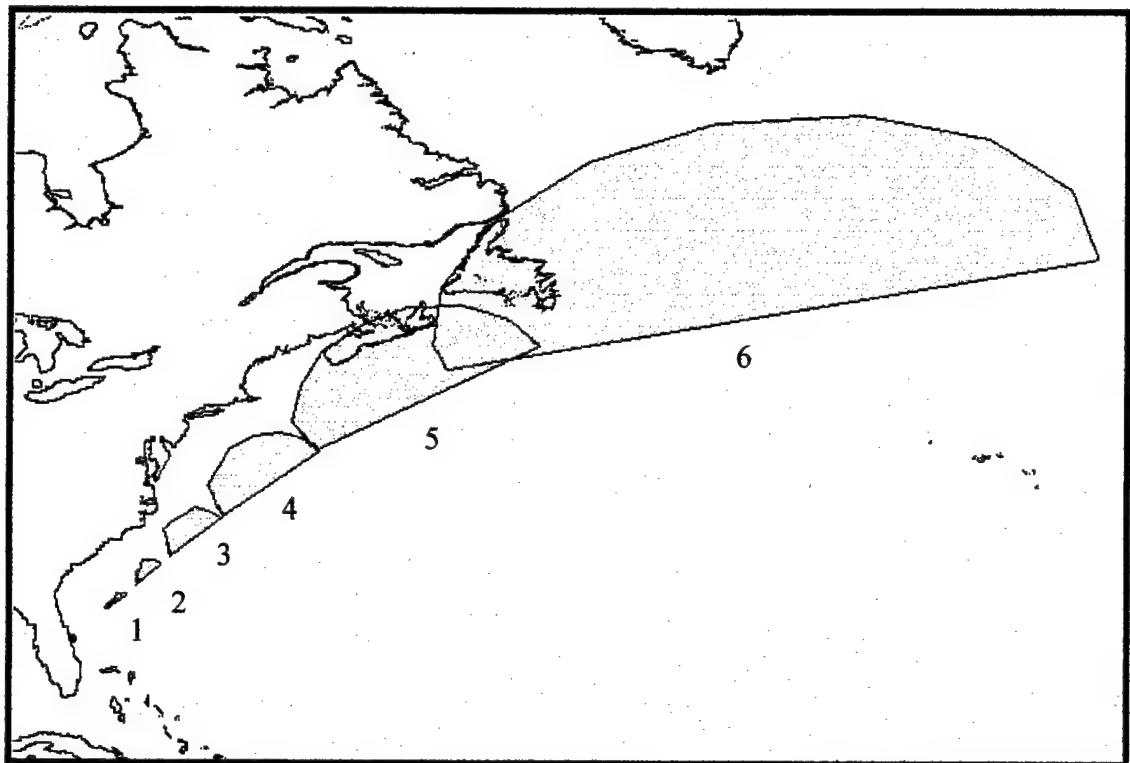
Figure 3-5. Abort Landing Zones for a 51.6 Degree Inclination Launch
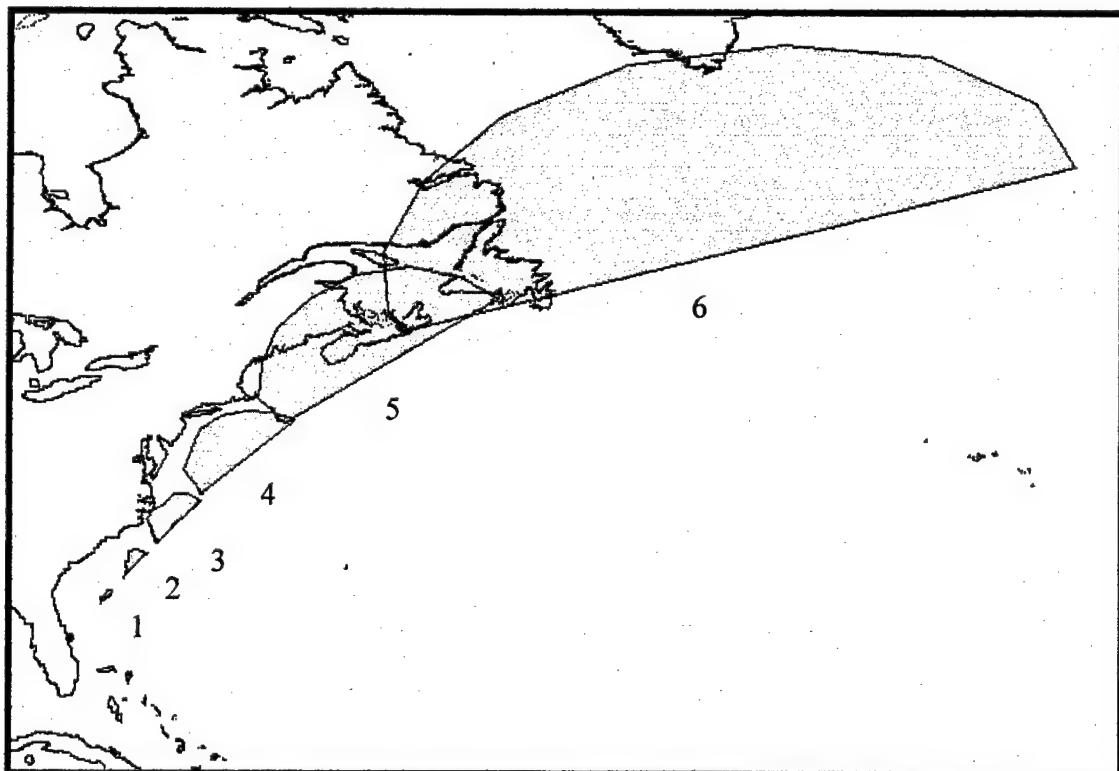
Figure 3-6. Abort Landing Zones for a 57 Degree Inclination Launch

These abort zones represent the possible locations of the Orbiter at 25,000 meters but don't display the speed. For the altitude of 25,000 meters, the nominal approach speed of the Orbiter is Mach 2.5 (6:1). The speeds of each combination were averaged to give an average velocity for each landing zone. Figures 3-7 and 3-8 show the average speed of each zone compared to the nominal approach speed.
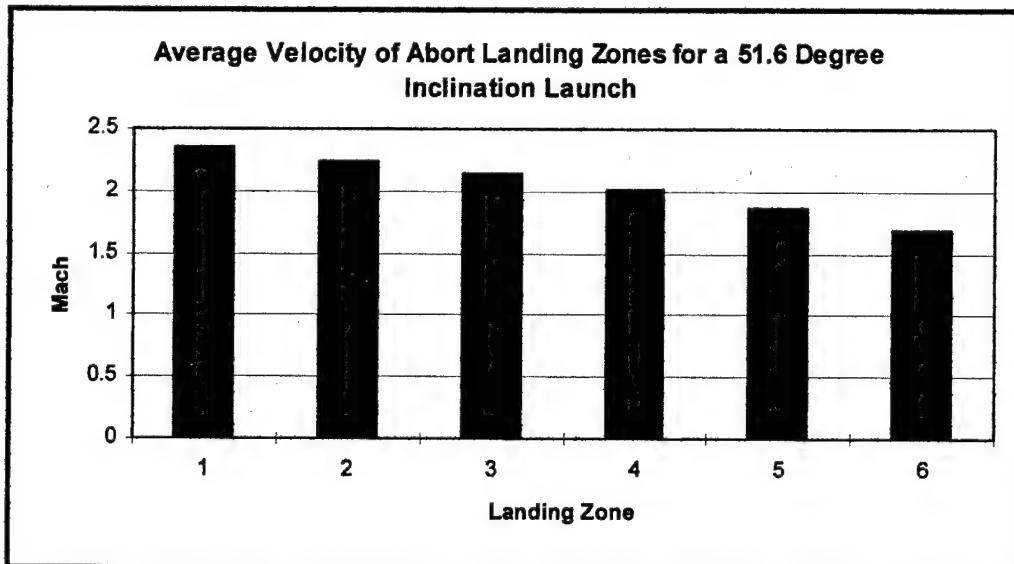
Figure 3-7. Average Velocities of Abort Landing Zones for a 51.6 Degree Inclination Launch
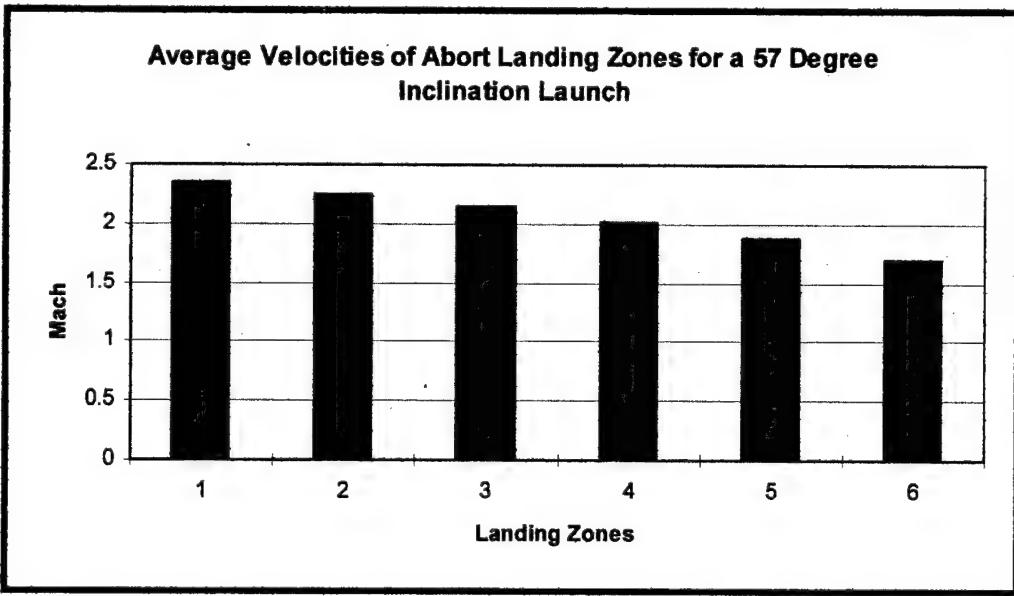
Figure 3-8. Average Velocities of Abort Landing Zones for a 57 Degree Inclination Launch

3.4 East Coast Airports

Table 3-1 lists thirty seven airports close to the launch path of the shuttle citing the location in longitude and latitude and giving runway lengths in feet .

Table 3-1. East Coast Airport Locations and Runway Lengths

Airport	Latitude	Longitude	Runway Length (ft)
Andrews AFB	38.82	-76.87	9755
Atlantic City Intl	39.45	-74.58	10000
Baltimore/Martin State	39.33	-76.42	6996
Bangor Intl	44.8	-68.83	11439
Beaufort MCAS	32.48	-80.72	12202
Brunswick NAS	43.9	-69.93	8000
Burlington Intl	44.47	-73.15	7821
Cecil Fld NAS	30.22	-81.88	12503
Charleston AFB	32.9	-80.03	9001
Cherry Point	34.9	-76.88	8980
Dover AFB	39.13	-75.47	12902
Gander	48.93	-54.57	10500
Goose Bay	53.02	-60.42	11000
Griffiss AFLD	43.23	-75.4	11820
Langley AFB	37.08	-76.35	10000
McEntire ANGS	33.92	-80.8	9001
McGuire AFB	40.02	-74.58	10001
Moncton	46.17	-64.57	8000
Narsarsuaq	61.17	-45.42	6000
Niagara Falls Intl	43.1	-78.95	9125
Norfolk NAS	36.93	-76.28	8369
Oceana NAS	36.82	-76.03	11997
Otis ANGB	41.65	-70.52	9500
Patuxent River NAS	38.28	-76.42	11800
Pope AFB	35.17	-79.02	7501
Richmond Intl	37.5	-77.32	8999
Savannah Intl	32.13	-81.2	9351
Seymour Johnson AFB	35.33	-77.97	11758
Shaw AFB	35.97	-80.47	10010
Sondrestrom AB	67.02	-50.72	9200
St. Johns	47.62	-52.75	8500
Sydney	46.17	-60	7100
Syracuse/Hancock Intl	43.12	-76.12	9003
West Field/Barnes Muni	42.17	-72.72	9000
Westover ARB	42.2	-72.53	11600
Willow Grove NAS	40.2	-75.15	8002
Windsor Locks/Bradley	41.95	-72.67	9502

In figures 3-9 and 3-10 these airport locations were plotted onto figures 3-5 and 3-6 to determine which would be the most likely abort landing sites. Primary landing sites will be those airports within the landing zone that have the longest runway while secondary sites will be within or close to the zone. It would be preferable to find runways of 10,000 ft. or more that could safely land the Orbiter using a drogue chute but if none are available a short runway is much better than ditching into the ocean. If no airports fall within the zone the closest airport will be chosen as a remote possibility. These remote sites would mean that the Orbiter would have to travel outside it's nominal landing approach and should be avoided if possible.
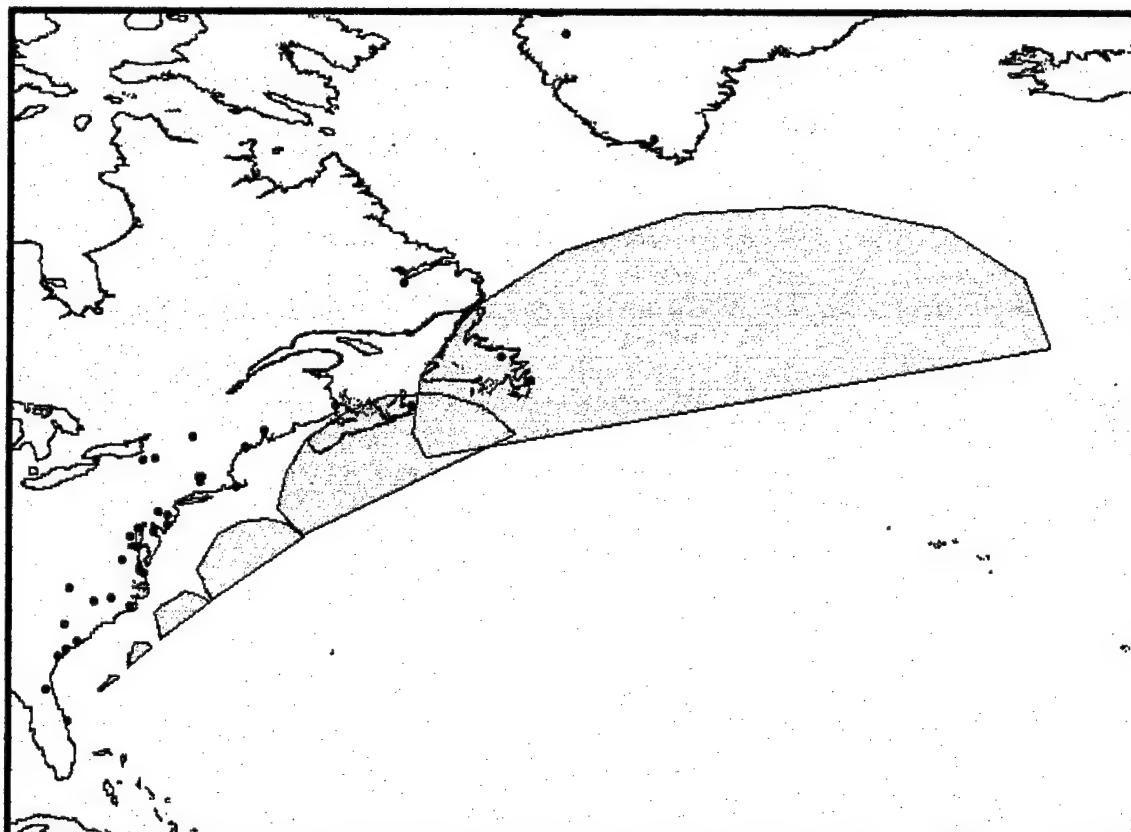
Figure 3-9. East Coast Airports Compared to 51.6 Degree Inclination Launch Abort Landing Zones
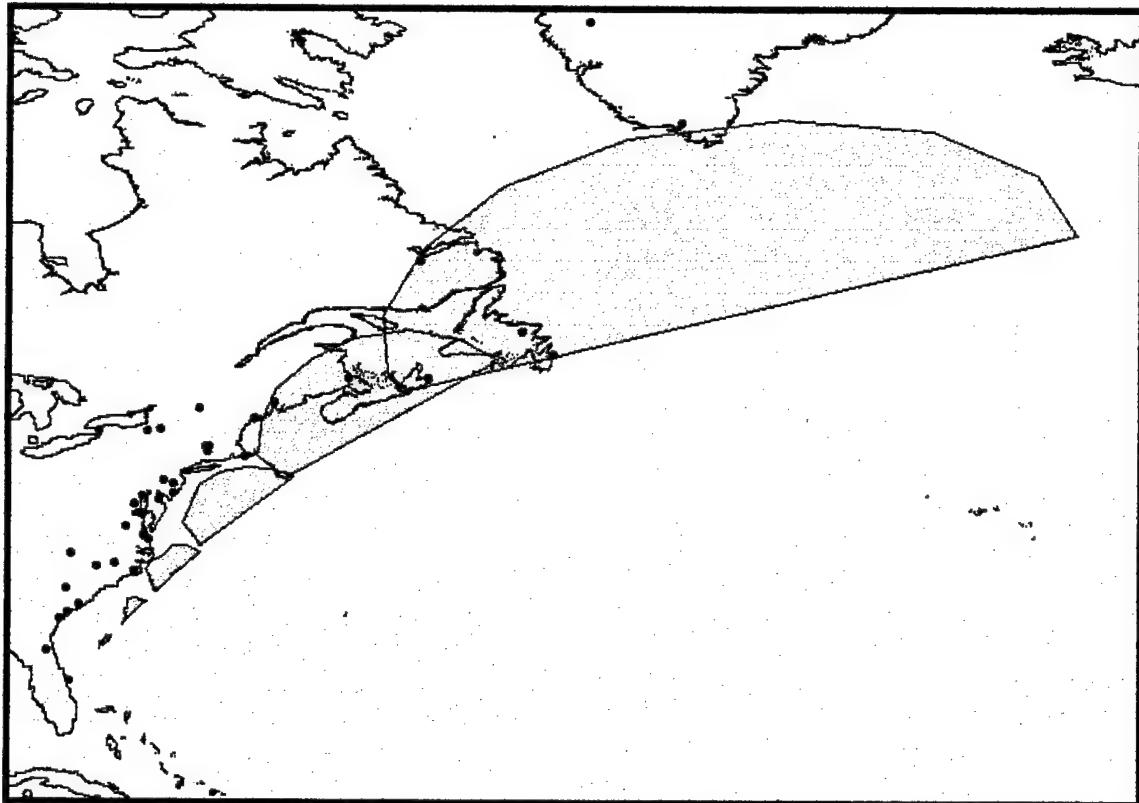
Figure 3-10. East Coast Airports Compared to 57 Degree Inclination Launch Abort Landing Zones

3.5 Results

Tables 3-2and 3-3 list possible abort landing sites for both 51.6 and 57 degree inclination launches using the guides established in the previous section.

Table 3-2. Abort Landing Sites for 51.6 Degree Inclination Launch

Landing Zone	No. of Airports within Zone	Primary Site	Secondary Site
1	None	** Charleston AFB	None
2	None	** Cherry Point	None
3	None	** Oceana NAS	None
4	None	**Ottis	None
5	1	Sydney	Moncton
6	2	Gander	St. Johns

Table 3-3. Abort Landing Sites for 57 Degree Inclination Launch

Landing Zone	No. of Airports within Zone	Primary Site	Secondary Site
1	None	** Charleston AFB	None
2	None	** Cherry Point	None
3	None	** Oceana NAS	None
4	None	**Ottis	None
5	3	Bangor	Moncton
6	4	Goose Bay	Gander

From the tables, it is plain to see that for both 51.6 and 57 degree inclination launches that the only airports that lie in the landing zones occur at 360 and 420 seconds into the launch. At these later times, the Orbiter is high enough that it could do an AOA or ATO and wouldn't need to abort to the east coast. The earlier abort points occurring at 120 and 180 seconds could abort using a RTLS so it isn't imperative to try a landing at Charleston or Cherry Point. The midrange landing zones however, have no alternative but to attempt the abort to Oceana and Ottis, which are far outside the predicted landing zones established by the Orbiter's nominal landing approach.

Furthermore, figures 3-7 and 3-8 indicate that the velocity of the orbiter in each zone is slower than the nominal mach 2.5 for the 25,000 meter altitude. This may cause the Orbiter to change it's glide slope
in order to make up for this reduction in energy. If the speed is too slow it may not make the runway at all and park the Orbiter in someone's back yard. In either case the landing zones do not represent a normal shuttle approach for landing.

The only conclusion that can be draw is that a constant control descent does not answer the question of east coast abort feasibility but only provides a guide for further control methods. The landing zones 2, 3, and 4 for both inclination launches come close to some airports whose runways could accommodate the Shuttle . These airports now become selected landing sites which the second control descent program, Abort2, will attempt to reach.

4. Data Analysis and Results for Controlled Descent to Landing Site

4.1 Introduction

The controlled descent method picks primary airports based on location and runway size and guides the orbiter to them. Pitch and roll only stay constant until the correct heading angle to intercept the primary airport is reached, then the roll angle becomes zero allowing the orbiter to glide straight toward the target. This is done for every point throughout the trajectory for both 51.6 and 57 degree inclination launches giving exact times when primary airports can be reached. Average velocity and altitude are given for all abort attempts within a zone around the target airport which is 1 degree of longitude by 1 degree of latitude with the runway in the center. These averages are then compared with the nominal landing criteria of a speed of mach 1and an estimated altitude of 6098 meters (6:1).

4.2 Controlled Descent Using Heading Angle

The Gravturn program once again simulates the launch of the shuttle at both 51.6 and 57 degrees but the abort program is slightly modified. Rather than cycling through all possible angles to create landing zones for six discrete points, an airport is chosen as a landing site and every point in the launch from 120 to 480 seconds at 2 second intervals is aborted to that site. This is done by choosing both pitch and roll as 45 degrees which are reasonable values that are within the flight envelope established in the Shuttle's aerodynamic flight model. These conditions remain constant until the proper heading angle to intercept the target airport is achieved at which point the roll and pitch angle become 0 and 30 degrees respectively.

Spherical trigonometry is used to calculate the proper heading angle to the prescribed airport knowing the latitude and longitude of the target as well as the latitude and longitude of the Orbiter at every point. The following two equations were derived from spherical trigonometry, because you must first solve one to obtain the other.

$$l = \cos^{-1}\left[\cos(\frac{\pi}{2} - \delta_{OR})\cos(\frac{\pi}{2} - \delta_{AP}) + \sin(\frac{\pi}{2} - \delta_{OR})\sin(\frac{\pi}{2} - \delta_{AP})\cos(\lambda_{OR} - \lambda_{AP})\right] \quad (4\text{-}1)$$

$$\psi_{AP} = \sin^{-1}\left[\frac{\sin(\frac{\pi}{2} - \delta_{AP})\sin(\lambda_{OR} - \lambda_{AP})}{\sin l}\right] + \frac{\pi}{2} \quad (4\text{-}2)$$

where

> l = *Radial Distance to Target* (radians)
>
> ψ_{AP} = *Heading Angle to Airport* (radians)
>
> δ_{AP} = *Latitude of Airport* (radians)
>
> δ_{OR} = *Latitude of Orbiter* (radians)
>
> λ_{AP} = *Longitude of Airport* (radians)
>
> λ_{OR} = *Longitude of Orbiter*(radians)

These two equations were placed in the Dynam subroutine of the Abort2 program so that each abort descent would roll to 0 degrees at the proper heading angle.

4.3 Abort Landing Sites

From the abort landing zones established in the constant control method certain airports were chosen as 'primary' and 'secondary' landing sites. Due to the lack of airports within the early abort regions several other airports were chosen as 'possible' landing sites if alternate control methods were used. For both the 51.6 and 57 degree inclination launches the same landing sites for the early aborts were chosen because neither launch produced landing zones which intercepted any airports. Since both launches have identical landing sites for the early abort points, the 57 degree launch is investigated first because it's trajectory brings it closer to the east coast and if the airports cannot be reached there is no point in doing the abort from the 51.6 degree launch.

The first three landing sites chosen were Charleston AFB, Cherry Point, and Oceana Naval Air Station (NAS). These sites represent the first three minutes of abort opportunity starting after SRB

separation at 120 seconds and ending at 300 seconds. Figures 4-1, 4-2, and 4-3 show the abort descents from the launch trajectory down to 6098 meters.
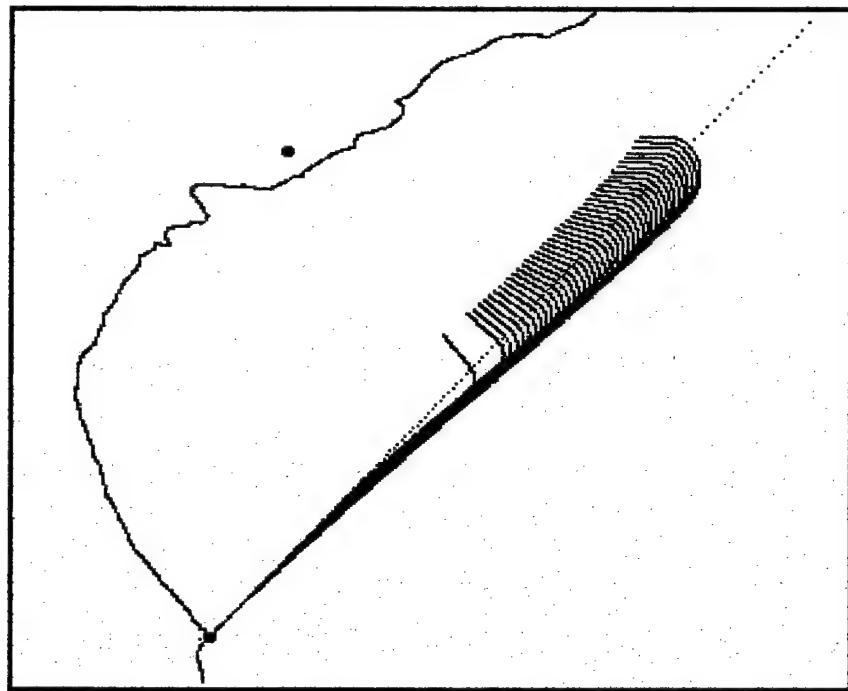
Figure 4-1. Abort Landing Attempt at Charleston AFB
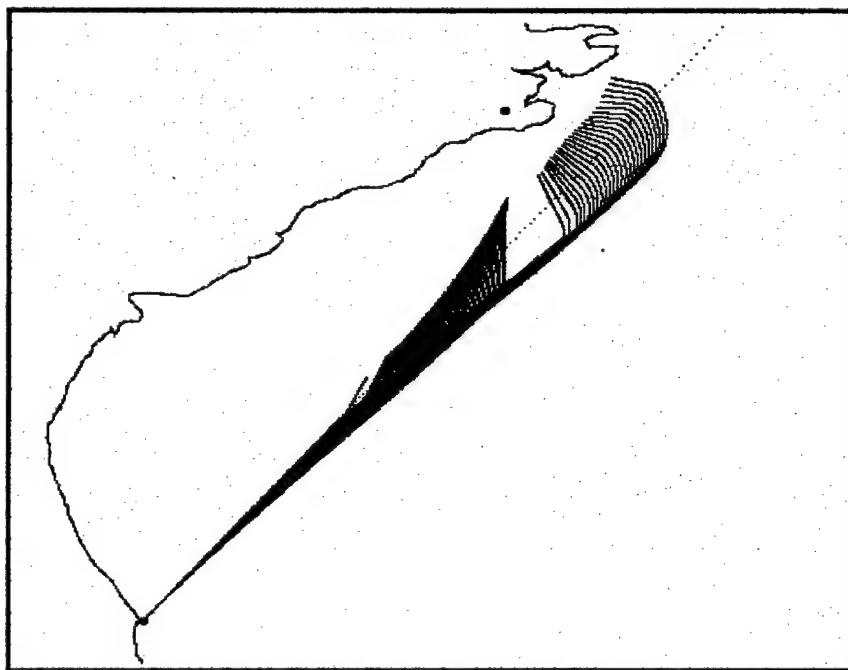
Figure 4-2. Abort Landing Attempt at Cherry Point
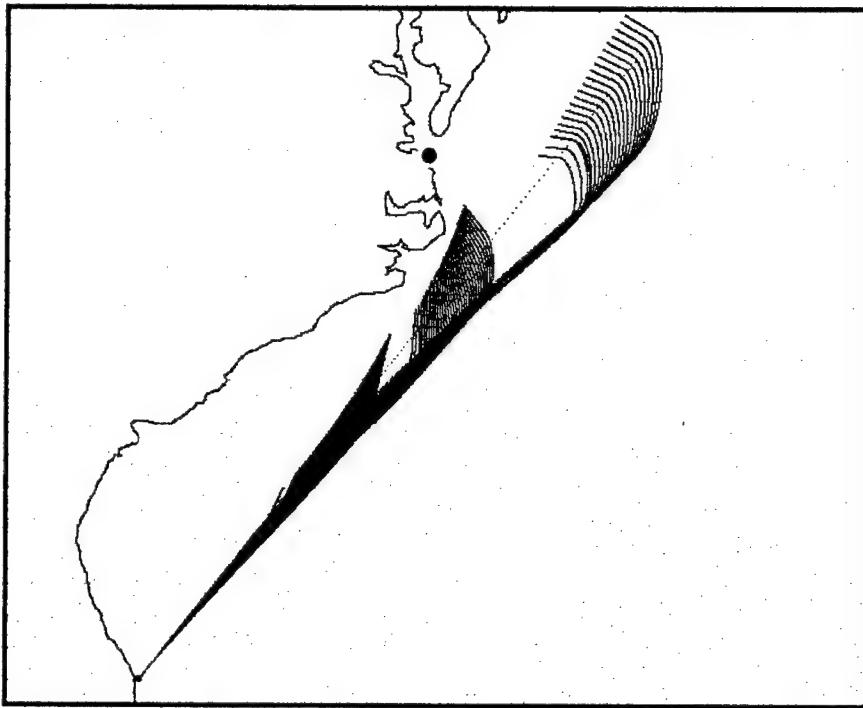
Figure 4-3. Abort Landing Attempt at Oceana NAS

These figures clearly illustrate that none of these three airports can be approached under nominal conditions using a non powered control descent from a 57 degree inclination launch with pitch and roll conditions as mentioned in section 4.2.
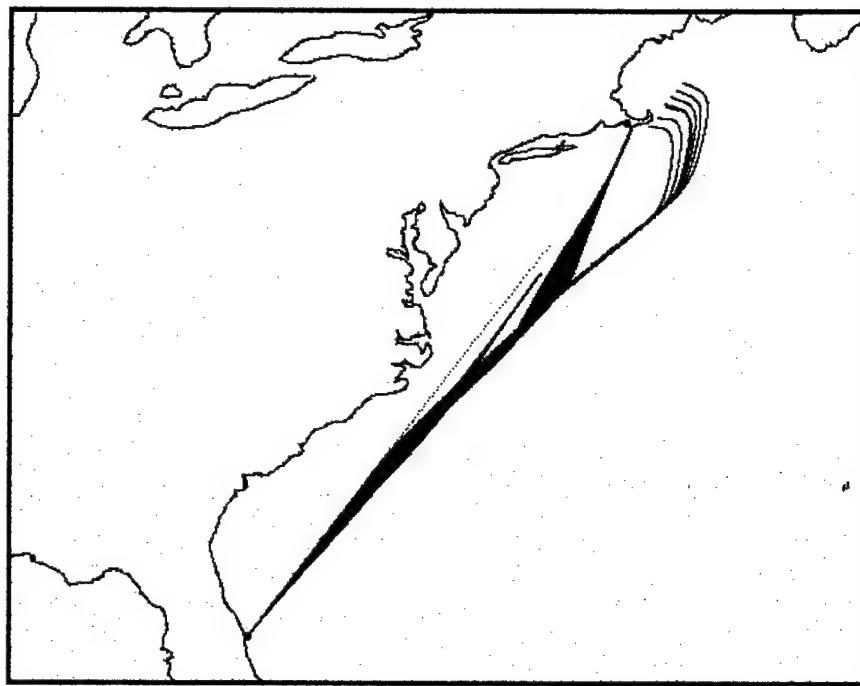
Figure 4-4. Abort Landing at Ottis ANGB for a 57 Degree Inclination Launch
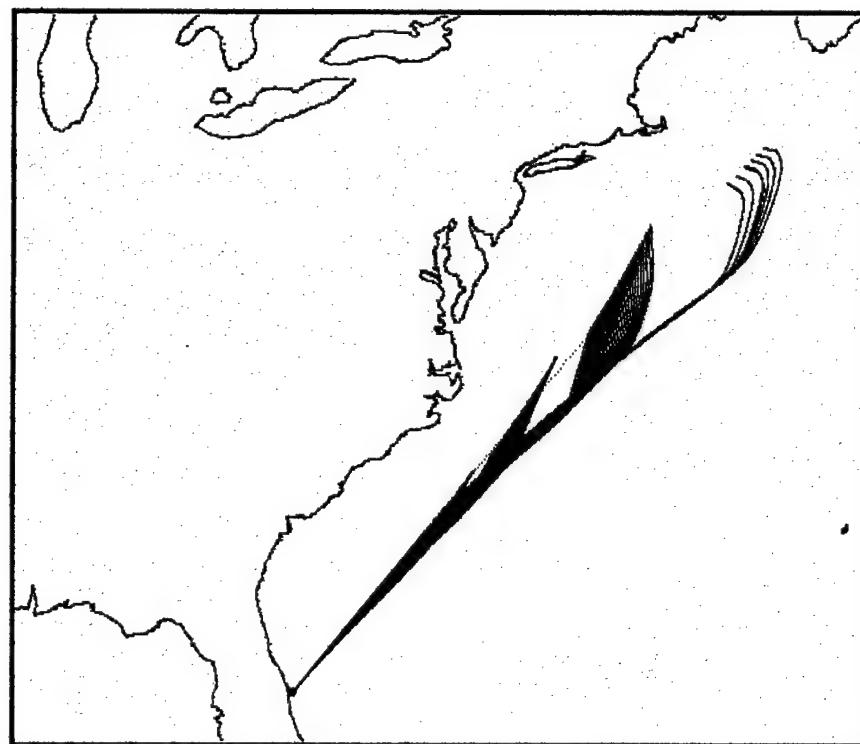
Figure 4-5. Abort Landing Attempt at Ottis ANGB for a 51.6 Degree Inclination Launch

The end of each abort represents the estimated nominal altitude of 6098 meters so it may be possible to reach these airports but the altitude would be too low to allow sufficient room to align with the runway and land safely. These results make running the abort descents for a 51.6 degree inclination unnecessary due to the longer distances that would need to be covered.

Ottis Air National Guard Base (ANGB) was the next landing site chosen for both 57 and 51.6 degree inclination launches. This site covers the abort opportunities from 300 to 360 seconds into the launch. Figures 4-4 shows that there are intercepts for the 57 degree launch but the window of opportunity is only 8 seconds at 330 seconds into the launch. Figure 4-5 shows that no intercepts are possible for the 51.6 degree inclination. These outcomes leave the 51.6 degree launch with no landing opportunities and the 57 degree launch with a very small one.

The next landing sites chosen were Sydney for the 51.6 degree launch and Moncton for the 57 degree inclination launch. The reason for choosing Moncton over Bangor was the result of reducing the flight envelope from 90 degree pitch to 45 degree pitch. This reduction doesn't allow the Orbiter to slow down fast enough to reach Bangor so the secondary site of Moncton was made the primary. Figures 4-6 and 4-7 illustrate these abort landings.
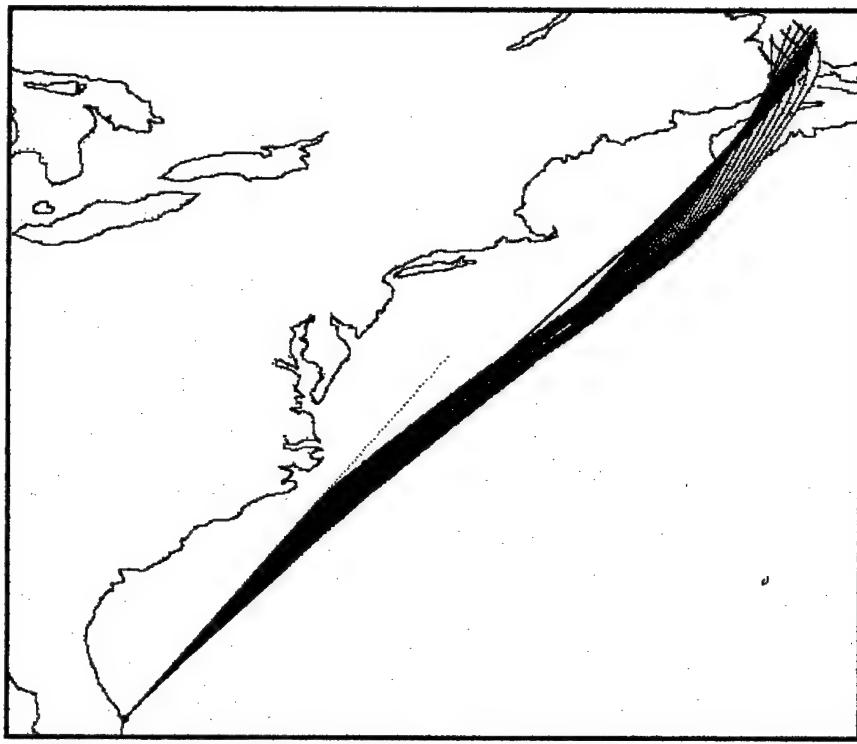
Figure 4-6. Abort Landing at Moncton for a 57 Degree Inclination Launch.
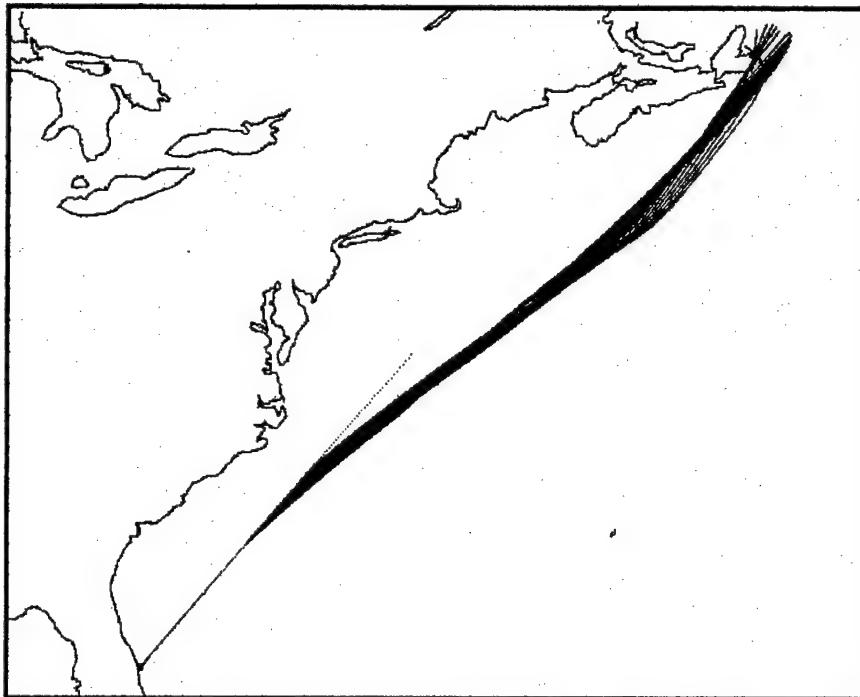
Figure 4-7. Abort Landing at Sydney for a 51.6 Degree Inclination Launch

Figure 4-6 proves that Moncton is within reach for many descents starting at 362 seconds into the launch up until 396 seconds. Figure 4-7 on the other hand shows that an abort to Sydney is possible but the window of opportunity is much smaller at 12 seconds starting at 390 seconds into the 51.6 degree inclination launch.

The final abort airports Gander, St. Johns and Goose Bay that represent 420 to 480 seconds into the launch were too close in the landing zone to be reachable using a 45 degree pitch. The remaining airports on the rim of the landing zones like Narsarsuaq and Sondrestrom were out of reach. Different configurations of pitch and roll which are out of the flight envelope may allow the Orbiter to reach these sites but velocity and altitude will be vital in determining whether the Orbiter can burn off enough energy to become slow enough and low enough to land.

4.4 Altitude and Velocity Profiles

The velocities and altitudes of all the descent trajectories within a 1 by 1 degree zone around the landing sites were averaged. Figures 4-8 and 4-9 compare these averages to the nominal approach velocity and altitude of mach 1 at 6098 meters. The speed of sound at 6098 meters was computed to be 347 m/s using the subroutine ATM and used as the nominal speed of approach.
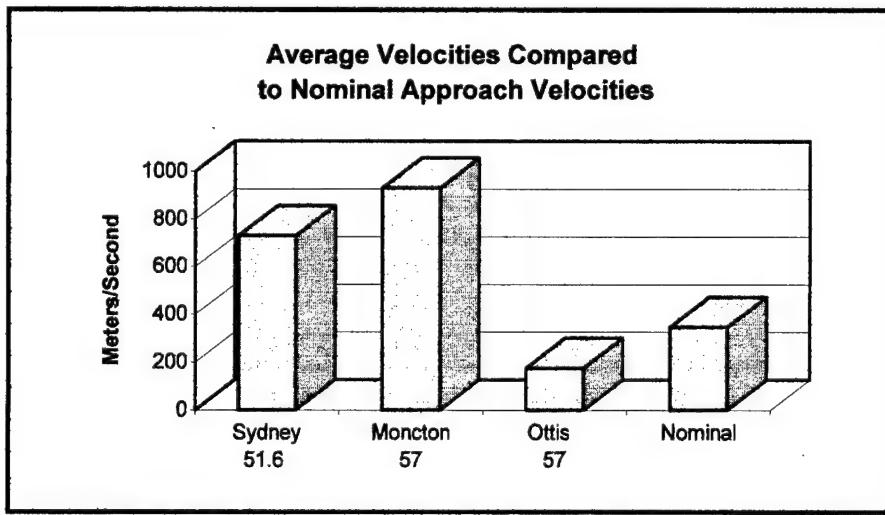
Figure 4-8. Average Velocities for All Airport Intercepts

4-8
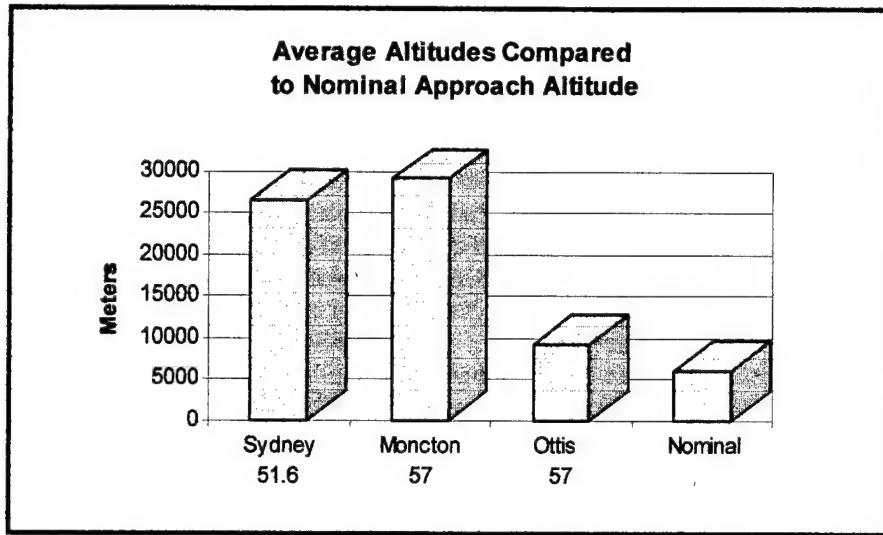
Figure 4-9. Average Altitudes for All Airport Intercepts

From these charts Sydney and Moncton are too high and too fast where as Ottis is too high and not fast enough. These results don't mean the Orbiter can't land at these sites but what they do suggest is that energy management techniques such as 'S' turns and high pitch angles may need to be employed to reduce speed and in turn altitude. To increase speed and reduce altitude a lower pitch angle might be employed .

4.5 Results and Recommendations

The method of controlled descent using heading angle for a 51.6 and 57 degrees inclination launch yield abort procedures outlined in table 4-1 and 4-2.

Table 4-1. Abort Procedures for a 51.6 Degree Inclination Launch

Time into Launch (seconds)	Abort Mode
120-280	RTLS
281-389	Contingency
390-402	ECA to Sydney
403 - up	Contingency, AOA, or ATO

Table 4-2. Abort Procedures for a 57 Degree Inclination Launch

Time into Launch (seconds)	Abort Mode
120-280	RTLS
281-329	Contingency
330-338	ECA to Ottis ANGB
339-361	Contingency
362-396	ECA to Moncton
397-up	Contingency, AOA, or ATO

As shown in both tables the Return To Launch Site (RTLS) abort mode must still be used up to 280 seconds into the launch. This is due to the fact that the Orbiter is not high or fast enough to reach any of the airports in the Southeast. For the 51.6 degree case the next 110 seconds would be a contingency or ditch type of abort as would the 57 degree case but only lasting 50 seconds. The 51.6 degree case can then East Coast Abort (ECA) to Sydney for 12 seconds after which depending on engine capability the Orbiter could ditch, Abort Once Around (AOA), or Abort To Orbit (ATO). The 57 degree case has two more opportunities to ECA to Ottis and Moncton at 330 and 362 respectively before having to either ditch, AOA, or ATO.

This investigation has shown that ECA's are possible but very limited. The early abort cases are still left up to the RTLS and contingency modes which are either extremely experimental or result in the loss of the Orbiter. Further study into ECAs using optimal controls may yield larger windows of opportunity for the abort landing sites mentioned as well as reduce speed and velocity to ensure safe landings. As for the early abort cases and the 51.6 degree launches, a powered case should also be investigated to maneuver the orbiter closer to the coast line before releasing the external tank . This may allow the 51.6 degree launches to reach the landing sites that the non-powered case could not as well as give both 57 and 51.6 launches the flexibility of choosing landing sites that may have longer runways and better facilities but were too far inland to reach.

In conclusion this thesis does indeed prove that East Coast Aborts are possible but more importantly proves that further research should be done to improve landing opportunities as well as increase the number of reachable airports in which to land. This study must also be followed up to investigate the aerodynamic moments the Shuttle may experience as a result of hypersonic maneuvers which were not looked at in this point mass model. In any case future research is the major recommendation of this paper because with the greater number of launches the shuttle makes the larger the chances that something could go wrong and to be able to save both crew and Orbiter would be beneficial. Not only does this benefit NASA but, if we can avoid a major catastrophe and prevent further budget cuts or an all out cancellation of the space program, then it would also benefit everyone who has ever looked up to the stars and dreamed.

Appendix A Gravity-Turn Trajectory Launch Program

Contained in this section is all the source code used in modeling the launch of the Space Shuttle.

The only subroutines that are omitted are the two Haming numerical integrators use to integrate Rhs2 and

Rhs3. They are left out because the code is the same except for which Rhs subroutine it calls to integrate.

```
        Program Gravturn
        implicit none
c
c     dynamics model for gravity-turn trajectory from Wiesel page 208 equations 67 - 78.
c
c     Input: Initial State ( Down Range Distance, Altitude, Velocity, Flight Path Angle,Mass)
c     Output: State vector at any time T ( X, H, V, Gamma)
c          (Printed to screen and selected files)
c
        common /ham/ t,x(230,4),f(230,4),err(230),n,h
        Real*8 t ,Long_Launch,Lat_Launch,Inclin,R
        Real*8 x , f , err , h , toler, Long , Lat, Psi
        integer nxt, n
        toler = 1.0d-9

c       File used in controlled Abort
        open (unit= 2,file = 'c:\afit\grav.out', status ='unknown')

c       File used in mapping program for ascent trajectory
        open (unit= 3,file = 'c:\afit\grav2.out',status ='unknown')

c       Files used in constant control abort and in the mapping program
        open (unit=4, file = 'c:\afit\abort1.out',status ='unknown')
        open (unit=8, file = 'c:\afit\abort2.out',status ='unknown')
        open (unit=9, file = 'c:\afit\abort3.out',status ='unknown')
        open (unit=10, file = 'c:\afit\abort4.out',status ='unknown')
        open (unit=11, file = 'c:\afit\abort5.out',status ='unknown')
        open (unit=12, file = 'c:\afit\abort6.out',status ='unknown')
```

```fortran
c               initialize(time,state) and constants (entered as SI units)
c
                x(1,1) = 0
                x(2,1) = 100
                x(3,1) = 32.1
                x(4,1) = 1.56209405
                x(5,1) = 2041200
                Long_Launch = -80.6
                Lat_Launch = 28.4583
                Inclin = 51.6
                R = 6378.135d3
                n =5
                t =0.d0
                h =2.0d0
                nxt =0
                call haming1(nxt)
                  if(nxt.eq.0)then
                    write(*,*) 'dead'
                    stop
                  endif
c                       First stage on launch profile
                do while(t.le.120.and.x(4,nxt).gt.toler)

                   call haming1(nxt)
c               Calculations for Long, Lat, and Psi

                   Lat = -dacosd(dcosd(90-Lat_Launch)*dcos(x(1,nxt)/R)
     1                  + dsind(90-Lat_Launch)*dsin(x(1,nxt)/R)
     2                  *dcosd(90-Inclin))+90
c
                   Long = dacosd((dcos(x(1,nxt)/R)-dcosd(90-Lat)
     1                  *dcosd(90-Lat_Launch))/(dsind(90-Lat)
     2                  *dsind(90-Lat_Launch)))+ Long_Launch
c
                   Psi =1.5708 - (dasin(dcosd(Inclin)/dcosd(Lat)))
c
                   write(3,*) Long,Lat
                   write(*,*)'state =', x(1,nxt),x(2,nxt),x(3,nxt), x(4,nxt),
     1                              x(5,nxt),t
                 if(t.eq.120)then
                     write(4,*) x(2,nxt)+R,Long, Lat,x(3,nxt),x(4,nxt),Psi
c                    For state at abort point 1
                     write(2,*) x(2,nxt)+R,Long, Lat,x(3,nxt),x(4,nxt),Psi
                 endif
                end do
c                       Initializing for second stage
                 x(1,1)= x(1,nxt)
                         x(2,1)= x(2,nxt)
                         x(3,1)= x(3,nxt)
                         x(4,1)= x(4,nxt)
                         x(5,1)= x(5,nxt)-164000
                           nxt= 0
```

A-2

```fortran
            call haming2(nxt)
              if(nxt.eq.0)then
                  write(*,*) 'dead2'
                  stop
              else
              write(*,*) 'alive'

              endif
c            second stage of launch profile
          do while(t.lt.480)

                call haming2(nxt)
c        Calculations for Long, Lat, and Psi
c
          Lat = -dacosd(dcosd(90-Lat_Launch)*dcos(x(1,nxt)/R)
     1          + dsind(90-Lat_Launch)*dsin(x(1,nxt)/R)
     2          *dcosd(90-Inclin))+90
c
          Long = dacosd((dcos(x(1,nxt)/R)-dcosd(90-Lat)
     1          *dcosd(90-Lat_Launch))/(dsind(90-Lat)
     2          *dsind(90-Lat_Launch)))+ Long_Launch
c
          Psi = 1.5708 - (dasin(dcosd(Inclin)/dcosd(Lat)))
c
          write(3,*) Long,Lat
          write(*,*)'state =', x(1,nxt),x(2,nxt),x(3,nxt),x(4,nxt),
     1              x(5,nxt),t
          write(2,*) x(2,nxt)+R,Long, Lat,x(3,nxt),x(4,nxt),Psi

c                  For abort point 2
          if(t.eq.180)then
            write(8,*) x(2,nxt)+R,Long, Lat,x(3,nxt),x(4,nxt),Psi
          endif
c          For abort point 3
          if(t.eq.240)then
            write(9,*) x(2,nxt)+R,Long, Lat,x(3,nxt),x(4,nxt),Psi
          endif
c          For abort point 4
          if(t.eq.300)then
            write(10,*) x(2,nxt)+R,Long, Lat,x(3,nxt),x(4,nxt),Psi
          endif
c          For abort point 5
          if(t.eq.360)then
            write(11,*) x(2,nxt)+R,Long, Lat,x(3,nxt),x(4,nxt),Psi
          endif
c          For abort point 6
          if(t.eq.420)then
            write(12,*) x(2,nxt)+R,Long, Lat,x(3,nxt),x(4,nxt),Psi
          endif

          end do
```

```
c           initialize for final stage
                    x(1,1)= x(1,nxt)
                    x(2,1)= x(2,nxt)
                    x(3,1)= x(3,nxt)
                    x(4,1)= x(4,nxt)
                    x(5,1)= x(5,nxt)-32000
                       nxt= 0

              call haming3(nxt)
          if(nxt.eq.0)then
              write(*,*) 'dead3'
              stop
          endif
c              Final stage
       do while(t.ge.480)
                 call haming3(nxt)
c       Calculations for Long, Lat, and Psi
              Lat = -dacosd(dcosd(90-Lat_Launch)*dcos(x(1,nxt)/R)
     1             + dsind(90-Lat_Launch)*dsin(x(1,nxt)/R)
     2             * dcosd(90-Inclin))+90
c
              Long = dacosd((dcos(x(1,nxt)/R)-dcosd(90-Lat)
     1               *dcosd(90-Lat_Launch))/(dsind(90-Lat)
     2               *dsind(90-Lat_Launch)))+ Long_Launch
c
              Psi = 1.5708 - (dasin(dcosd(Inclin)/dcosd(Lat)))
c
          write(3,*) Long,Lat
          write(2,*) x(2,nxt)+R,Long, Lat,x(3,nxt),x(4,nxt),Psi
          write(*,*)'state =', x(1,nxt),x(2,nxt),x(3,nxt),x(4,nxt),
     1               x(5,nxt),t
          stop
       end do

      write(*,*) 'Finished Successfully with'
      write(*,*) ' Gamma = ', x(4,nxt), 'at time',t ,nxt
        close(2)
        close(3)
        close(4)
        close(8)
        close(9)
        close(10)
        close(11)
        close(12)
      stop

   end
```

Subroutine Rhs1(nxt)

```fortran
c
c       Equations of Motion for gravity-turn trajectory from Wiesel page 208 equations 67 - 78.
c       covering from t=0s to t=120s
c
c
c        RHS needs access to the common /ham/:
         common /ham/ t,x(230,4), f(230,4),err(230),n,h,mode
         real*8 t,x,f,err,h,mu,R,m_dot1,Tr,P0,PALT,TALT,DALT,dDdr
         real*8 CDA1
c        List of Constants
         R =6378.135d3
         mu =3.98601d14
         m_dot1 =9906
         Tr =30300d3
         P0 = 101325.0d0
         CDA1 = 101.12




c        Must include the gravity term
c
         g = mu/((x(2,nxt)+ R)*(x(2,nxt)+ R))
c
c        Equations of Motion from Wiesel
c
c    f(1) - d(x)/dt - Down Range Distance
c    f(2) - d(H)/dt - Altitude
c    f(3) - d(V)/dt - Velocity
c    f(4) - d(gamma)/dt - Flight Path Angle
c    f(5) - d(mass)/dt - Mass
c
c    Wiesel Equations 67-70 (including Drag)
c
         f(1,nxt)= x(3,nxt)*dcos(x(4,nxt))

         f(2,nxt)= x(3,nxt)*dsin(x(4,nxt))

c        Calling subroutine ATM to get Density at Altitude
         call ATM(x(2,nxt),P0,PALT,TALT,DALT,dDdr)

         f(3,nxt)= (Tr/x(5,nxt)) - (g*dsin(x(4,nxt)))
1               + ((f(1,nxt)*f(1,nxt))/(x(2,nxt)+R))*dsin(x(4,nxt))
2               - ((CDA1*.5*DALT*x(3,nxt)*x(3,nxt))/x(5,nxt))

         f(4,nxt)= -(g*dcos(x(4,nxt))/(x(3,nxt)))
1               + (f(1,nxt)*f(1,nxt)*dcos(x(4,nxt))/(x(3,nxt)*(x(2,nxt)+R)))

         f(5,nxt)= -m_dot1

         return
         end
```

Subroutine Rhs2(nxt)

```
c
c       Equations of Motion for gravity-turn trajectory from Wiesel page 208 equations 67 - 78.
c       For second stage from 120s to 480s
c
c
c          RHS needs access to the common /ham/:
           common /ham/ t,x(230,4), f(230,4),err(230),n,h,mode
           real*8 t,x,f,err,h,mu,R,m_dot2,Tr2,P0,PALT,TALT,DALT,dDdr


           R =6378.135d3
           mu =3.98601d14
           m_dot2 =1496.0d0
c          full Thrust
c          Tr2 =6300d3
c          partial Thrust
           Tr2 = 6000d3

           CDA2 = 79.5
           P0 = 101325.0d0



c
c          Must include the gravity term
c
           g = mu/((x(2,nxt)+ R)*(x(2,nxt)+ R))
c
c          Equations of Motion from Wiesel
c
c    f(1) - d(x)/dt - Down Range Distance
c    f(2) - d(H)/dt - Altitude
c    f(3) - d(V)/dt - Velocity
c    f(4) - d(gamma)/dt - Flight Path Angle
c    f(5) - d(mass)/dt - Mass
c
c    Wiesel Equations 67-70 (Including Drag)
c
           f(1,nxt)= x(3,nxt)*dcos(x(4,nxt))
           f(2,nxt)= x(3,nxt)*dsin(x(4,nxt))
c          Calling subroutine ATM to get Density at Altitude
           call ATM(x(2,nxt),P0,PALT,TALT,DALT,dDdr)

           f(3,nxt)= (Tr2/x(5,nxt)) - (g*dsin(x(4,nxt)))
     1              + ((f(1,nxt)*f(1,nxt))/(x(2,nxt)+R))*dsin(x(4,nxt))
     2              - ((CDA2*.5*DALT*x(3,nxt)*x(3,nxt))/x(5,nxt))
```

```
c         To control shape of ascent by positive input to gamma
          if(t.eq.210)then
          x(4,nxt) = x(4,nxt) + .11
          endif

          if(t.eq.280)then
          x(4,nxt) = x(4,nxt) + .12
          endif

          if(t.eq.340)then
          x(4,nxt) = x(4,nxt) + .10
          endif

          if(t.eq.420)then
          x(4,nxt) = x(4,nxt) + .02
          endif

          f(4,nxt)= -(g*dcos(x(4,nxt))/(x(3,nxt)))
1             + (f(1,nxt)*f(1,nxt)*dcos(x(4,nxt))/(x(3,nxt)*(x(2,nxt)+R)))

          f(5,nxt)= -m_dot2

          return
          end
```

Subroutine Rhs3(nxt)

```
c
c       Equations of Motion for gravity-turn trajectory from Wiesel page 208 equations 67 - 78.
c       For final stage for over 480s
c
c
c           RHS needs access to the common /ham/:
            common /ham/ t,x(230,4), f(230,4),err(230),n,h,mode
            real*8 t,x,f,err,h,mu,R,m_dot3,Tr3,P0,PALT,TALT,DALT,dDdr


            R =6378.135d3
            mu =3.98601d14
            m_dot3 =0
            Tr3 =0
            P0 = 101325.0d0
            CDA3 = 29.235



c           Must include the gravity term
c
            g = mu/((x(2,nxt)+ R)*(x(2,nxt)+ R))
c
c           Equations of Motion from Wiesel
c
c   f(1) - d(x)/dt - Down Range Distance
c   f(2) - d(H)/dt - Altitude
c   f(3) - d(V)/dt - Velocity
c   f(4) - d(gamma)/dt - Flight Path Angle
c   f(5) - d(mass)/dt - Mass
c
c   Wiesel Equations 67-70 (Including Drag)
c
            f(1,nxt)= x(3,nxt)*dcos(x(4,nxt))
            f(2,nxt)= x(3,nxt)*dsin(x(4,nxt))
c           Calling subroutine ATM to get Density at Altitude
            call ATM(x(2,nxt),P0,PALT,TALT,DALT,dDdr)
c
            f(3,nxt)= (Tr3/x(5,nxt)) - (g*dsin(x(4,nxt)))
     1              + ((f(1,nxt)*f(1,nxt))/(x(2,nxt)+R))*dsin(x(4,nxt))
     2              - ((CDA3*.5*DALT*x(3,nxt)*x(3,nxt))/x(5,nxt))
            f(4,nxt)= -(g*dcos(x(4,nxt))/(x(3,nxt)))
     1              + (f(1,nxt)*f(1,nxt)*dcos(x(4,nxt))/(x(3,nxt)*(x(2,nxt)+R)))
            f(5,nxt)= -m_dot3

            return
            end
```

Subroutine Haming1(nxt)

```
c
c     haming is an ordinary differential equations integrator
c     it is a fourth order predictor-corrector algorithm
c     which means that it carries along the last four
c     values of the state vector, and extrapolates these
c     values to obtain the next value (the prediction part)
c     and then corrects the extrapolated value to find a
c     new value for the state vector.
c
c     the value nxt in the call specifies which of the 4 values
c     of the state vector is the "next" one.
c     nxt is updated by haming automatically, and is zero on
c     the first call
c
c     the user supplies an external routine rhs(nxt) which
c     evaluates the equations of motion
c
      common /ham/ x,y(230,4),f(230,4),errest(230),n,h,mode
      double precision x,y,f,errest,h,hh,xo,tol,eps
c
c     all of the good stuff is in this common block.
c     x is the independent variable ( time )
c     y(6,4) is the state vector- 4 copies of it, with nxt
c            pointing at the next one
c     f(6,4) are the equations of motion, again four copies
c            a call to rhs(nxt) updates an entry in f
c     errest is an estimate of the truncation error - normally not
c            used
c     n is the number of equations being integrated - 6 or 42 here
c     h is the time step
c     mode is 0 for just EOM, 1 for both EOM and EOV
c
      tol = 0.00000001d+00
c     switch on starting algorithm or normal propagation
      if(nxt) 190,10,200
c
c     this is hamings starting algorithm....a predictor - corrector
c     needs 4 values of the state vector, and you only have one- the
c     initial conditions.
c     haming uses a Picard iteration (slow and painfull) to get the
c     other three.
c     if it fails, nxt will still be zero upon exit, otherwise
c     nxt will be 1, and you are all set to go
c
   10 xo = x
      hh = h/2.0d+00
      call rhs1(1)
      do 40 l = 2,4
```

```fortran
      x = x + hh
      do 20 i = 1,n
 20   y(i,l) = y(i,l-1) + hh*f(i,l-1)
      call rhs1(l)
      x = x + hh
      do 30 i = 1,n
 30   y(i,l) = y(i,l-1) + h*f(i,l)
 40   call rhs1(l)
      jsw = -20
 50   isw = 1
      do 120 i = 1,n
      hh = y(i,1) + h*( 9.0d+00*f(i,1) + 19.0d+00*f(i,2)
     1    - 5.0d+00*f(i,3) + f(i,4) ) / 24.0d+00
      if(y(i,2) .eq. 0.d0) then
        eps = dabs( hh - y(i,2) )
      else
        eps = dabs( (hh - y(i,2))/y(i,2) )
      endif
      if( eps .lt. tol ) go to 70
      isw = 0
 70   y(i,2) = hh
      hh = y(i,1) + h*( f(i,1) + 4.0d+00*f(i,2) + f(i,3))/3.0d+00
      if(y(i,3) .eq. 0.d0) then
        eps = dabs( hh - y(i,3) )
      else
        eps = dabs( (hh-y(i,3))/y(i,3) )
      endif
      if( eps .lt. tol ) go to 90
      isw = 0
 90   y(i,3) = hh
      hh = y(i,1) + h*( 3.0d+00*f(i,1) + 9.0d+00*f(i,2) + 9.0d+00*f(i,3)
     1    + 3.0d+00*f(i,4) ) / 8.0d+00
      if(y(i,4) .eq. 0.d0) then
        eps = dabs( hh - y(i,4) )
      else
        eps = dabs( (hh-y(i,4))/y(i,4) )
      endif
      if( eps .lt. tol ) go to 110
      isw = 0
 110  y(i,4) = hh
 120  continue
      x = xo
      do 130 l = 2,4
      x = x + h
 130  call rhs1(l)
      if(isw) 140,140,150
 140  jsw = jsw + 1
      if(jsw) 50,280,280
 150  x = xo
      isw = 1
      jsw = 1
      do 160 i = 1,n
 160  errest(i) = 0.0
```

```fortran
      nxt = 1
      go to 280
  190 jsw = 2
      nxt = iabs(nxt)
c     this is hamings normal propagation loop -
c
  200 x = x + h
      np1 = mod(nxt,4) + 1
      go to (210,230),isw
c     permute the index nxt modulo 4
  210 go to (270,270,270,220),nxt
  220 isw = 2
  230 nm2 = mod(np1,4) + 1
      nm1 = mod(nm2,4) + 1
      npo = mod(nm1,4) + 1
c
c     this is the predictor part
c
      do 240 i = 1,n
      f(i,nm2) = y(i,np1) + 4.0d+00*h*( 2.0d+00*f(i,npo) - f(i,nm1)
     1    + 2.0d+00*f(i,nm2) ) / 3.0d+00
  240 y(i,np1) = f(i,nm2) - 0.925619835d0*errest(i)
c
c     now the corrector - fix up the extrapolated state
c     based on the better value of the equations of motion
c
      call rhs1(np1)
      do 250 i = 1,n
      y(i,np1) = ( 9.0d+00*y(i,npo) - y(i,nm2) + 3.0d+00*h*( f(i,np1)
     1    + 2.0d+00*f(i,npo) - f(i,nm1) ) ) / 8.0d+00
      errest(i) = f(i,nm2) - y(i,np1)
  250 y(i,np1) = y(i,np1) + 0.0743801653d0 * errest(i)
      go to (260,270),jsw
  260 call rhs1(np1)
  270 nxt = np1
  280 return
      end
```

```
Subroutine Atm(alt,p0,palt,talt,dalt,dddr)
c
c    earth atmosphere program, regan and anandarskarian, aiaa
c    "dynamics of atmospheric re-entry", appendix a
c
c    input:   alt    altitude in meters
c             po     ground level pressure, n/sq m
c    output:  palt   pressure at altitude, n/sq m
c             talt   temperature at altutude, deg c
c             dalt   density at altitude, kg/cu m
c             dddr   density gradient, kg/m^4
c
      double precision z(21),tm(21),lr(21),b,g0,r,d(21),p(21),p0,d0,rr
      double precision  talt,palt,dalt,alt,m(21),dd(21)
      double precision  e1,e2,e3,e4,e5,re,galt
      double precision dddr,de1dr,de2dr,de3dr
c
c    data stmts for break altitudes, temperatures, and molecular wts
c
c    altitudes
      data (z(i),i=1,21)/ 0.d3,  11.0191d3,  20.0631d3,  32.1619d3,
     1          47.3501d3,  51.4125d3,  71.8020d3,  86.00d3,
     2          100.d3,    110.d3,    120.d3,    150.d3,
     3          160.d3,    170.d3,    190.d3,    230.d3,
     4          300.d3,    400.d3,    500.d3,    600.d3,
     5          700.d3 /
c    molecular temperature
      data (tm(i),i=1,21)/ 300.d0,  216.65d0,  216.65d0,  228.65d0,
     1          270.65d0,  270.65d0,  214.65d0,  186.946d0,
     2          210.65d0,  260.65d0,  360.65d0,  960.65d0,
     3          1110.60d0, 1210.65d0, 1350.65d0, 1550.65d0,
     4          1830.65d0, 2160.65d0, 2420.65d0, 2590.65d0,
     5          2700.0d0 /
c    molecular wts
      data (m(i),i=1,21)/ 28.9664d0,  28.964d0,  28.964d0,  28.964d0,
     1          28.964d0,  28.964d0,  28.962d0,  28.962d0,
     2          28.880d0,  28.560d0,  28.070d0,  26.920d0,
     3          26.660d0,  26.500d0,  25.850d0,  24.690d0,
     4          22.660d0,  19.940d0,  17.940d0,  16.840d0,
     5          16.170d0 /
c    first pass flag
      data ifirst / 0 /
c
c    define constants on first pass
c
      if( ifirst .ne. 0 ) go to 1000
c
      b=3.139d-7
c    acceleration of gravity
```

A-12

```fortran
      g0=9.7803d0
c     universal gas const, j/kg
      rr=8313.432d0
      r=rr/m(1)
c     planetary radius, meters
      re = 6378145d0
c
c     initialize lapse rate for altitude regions
c
      do 10, l=1,21
         lr(l) = ( tm(l+1) - tm(l) )/( z(l+1) - z(l) )
10    continue
69    close(7)
      d0=p0/(r*tm(1))
      p(1)=p0
      d(1)=d0
      do 20,l=1,20
         r=rr/m(l)
         if (lr(l).eq.0.d0) then
              e1=1.d0-(b/2.d0)*(z(l+1)-z(l))
              e2=g0*(z(l+1)-z(l))/(r*tm(l))
              p(l+1)=p(l)*dexp(-e1*e2)
              d(l+1)=d(l)*dexp(-e1*e2)
              dd(l)=d(l+1)-d(l)/(z(l+1)-z(l))

         elseif (lr(l).ne.0.d0) then
              e1=1.d0+(lr(l)/tm(l))*(z(l+1)-z(l))
              e2=g0*b/(r*lr(l))
              e3=e2*(z(l+1)-z(l))
              e4=e2/b*(b/e2+1.d0+b*((tm(l)/lr(l))-z(l)))
              e5=e2/b*(1.d0+ b*((tm(l)/lr(l))-z(l)))
              p(l+1)=p(l)*(e1**(-e5))*dexp(e3)
              d(l+1)=d(l)*(e1**(-e4))*dexp(e3)
              dd(l)=(d(l+1)-d(l))/(z(l+1)-z(l))
         endif
20    continue
c
      ifirst = 1
c
1000  continue
c
c     determine which region altitude falls into
c
      do 500 j = 1,20
         if( alt .lt. z(j+1) ) then
             i = j
             go to 501
         endif
500   continue
      i = 21
501   continue
c
c     determine parameters at altitude
```

```fortran
c
      talt=tm(i)+lr(i)*(alt-z(i))
      galt=g0*(re**2/((re+alt)**2))
        r=rr/m(i)
c    if lapse rate is zero
c
      if (lr(i).eq.0.d0) then
          e1=1.d0-(b/2.d0)*(alt-z(i))
          e2=g0*(alt-z(i))/(r*tm(i))
          palt=p(i)*dexp(-1.d0*e1*e2)
          dalt=d(i)*dexp(-1.d0*e1*e2)
          de1dr = -b/2.d0
          de2dr = g0/(r*tm(i))
          dddr = -d(i)*dexp(-e1*e2)*( e1*de2dr + de1dr*e2)
c
c    if lapse rate not equal to zero
c
      elseif (lr(i).ne.0.d0) then
          e1=1.d0+(lr(i)/(tm(i)))*(alt-z(i))
          e2=g0*b/(r*lr(i))
          e3=e2*(alt-z(i))
          e4=e2/b*(b/e2+1.d0+b*((tm(i)/lr(i))-z(i)))
          e5=e2/b*(1.d0+ b*((tm(i)/lr(i))-z(i)))
          palt=p(i)*(e1**(-e5))*dexp(e3)
          dalt=d(i)*(e1**(-e4))*dexp(e3)
          de1dr = lr(i)/tm(i)
          de3dr = e2
          dddr = d(i)*( -e4*(e1**(-e4-1.d0))*de1dr
     1         + (e1**(-e4))*de3dr )*dexp( e3 )
      endif
c
      return
      end
```

A-14

Appendix B Abort Descent Program

This section contains all the source code for both methods of descent. The Abort1 program models the constant controlled descent while Abort2 covers the heading angle controlled descent. The only subroutines not included are Haming and Atm which are also used in the Gravturn program in Appendix A. The subroutine Dynam although included in this section only once has a portion of the code that was only used in the Abort2 program.

```
        Program Abort1
        implicit Real*8 (a-h,o-z)
c
c    3D reentry over a spherical rotating earth.  Vinh pp 21-27
c    assuming controls are pitch and roll which are held constant for each run
c     but cycled through all possible angular configurations
c
c    input: state is radius r, longitude, latitude, speed, flight path
c    angle, and heading angle
c
c        output: radius, longitude, latitude, speed, time
        common/ham/t,x(230,4),f(230,4),err(230),n,h
c
        common/param/Lalt,pitch,roll,p0,mu,R,M,S,Omega, MAC
        real*8    Lalt,t,h,x,f,err,pitch,roll,p0,mu,R,M,S
        real*8  Omega, MAC
        integer nxt,n

c        Files to store the abort landing Zones
        open(unit= 13,file='c:\afit\land1.out',status='unknown')
        open(unit= 14,file='c:\afit\land2.out',status='unknown')
        open(unit= 15,file='c:\afit\land3.out',status='unknown')
        open(unit= 16,file='c:\afit\land4.out',status='unknown')
        open(unit= 17,file='c:\afit\land5.out',status='unknown')
        open(unit= 18,file='c:\afit\land6.out',status='unknown')

c        Files to store the velocity data at nominal landing
        open(unit= 19,file='c:\afit\Vel1.out',status='unknown')
        open(unit= 20,file='c:\afit\Vel2.out',status='unknown')
        open(unit= 21,file='c:\afit\Vel3.out',status='unknown')
        open(unit= 22,file='c:\afit\Vel4.out',status='unknown')
        open(unit= 23,file='c:\afit\Vel5.out',status='unknown')
        open(unit= 24,file='c:\afit\Vel6.out',status='unknown')
```

```fortran
c      constants: (in SI Units)
       mu = 3.98601d14
       R =  6378.135d3
       p0 = 101325.0d0
       M =  108864.0d0
       S =  249.90d0
       Omega = 0.000072921158560d0
       MAC = 12.0580d0
       Lalt = 25000


c         This is the loop which cycles through all angles by an increment of 5 degrees

          do 1000 i = 30, 90, 5
          pitch = i/57.29578
          write(*,*) pitch
          do 2000 j = 0,90,5
          roll = j/57.29578
          write(*,*) roll
c         open each state vector produced by gravity turn program one at a time

          open (unit=4, file = 'c:\afit\abort1.out',status ='unknown')
c         open (unit=8, file = 'c:\afit\abort2.out',status ='unknown')
c         open (unit=9, file = 'c:\afit\abort3.out',status ='unknown')
c         open (unit=10, file = 'c:\afit\abort4.out',status ='unknown')
c         open (unit=11, file = 'c:\afit\abort5.out',status ='unknown')
c         open (unit=12, file = 'c:\afit\abort6.out',status ='unknown')
c         read in state from gravity turn program to be used as initial state
c         (radius,longitude, latitude,velocity,flight path angle, heading angle)

          read(4,*) x(1,1),x(2,1),x(3,1),x(4,1),x(5,1),x(6,1)
          x(2,1)= x(2,1)/57.29578
          x(3,1)= x(3,1)/57.29578
          n = 6
          h = .05d0
          t = 0.d0
          nxt = 0
         call haming(nxt)
          if(nxt.eq.0)then
            write(*,*) 'dead'
            stop
          endif
         do while(x(1,nxt)-R.ge.Lalt)
          call haming(nxt)
c         write to the correct file for each abort run
            write(13,*) x(2,nxt)*57.29578,x(3,nxt)*57.29578
            write(19,*)      x(4,nxt),t
            endif
         close(4)
         end do
          2000  continue
          1000  continue
         stop
         end
```

Program Abort2

```
        implicit Real*8 (a-h,o-z)
c
c    3D reentry over a spherical rotating earth.  Vinh pp 21-27
c    assuming controls are 45 degrees of pitch and roll until the proper heading
c    angle to intercept the landing site is achieved then pitch = 30 and roll = 0 degrees
c
c    input: state is radius r, longitude, latitude, speed, flight path
c    angle, and heading angle
c
c        output:  radius, longitude, latitude, speed, time

        common/ham/t,x(230,4),f(230,4),err(230),n,hc
        common/param/Lalt,pitch,roll,p0,mu,R,M,S,Omega,MAC,
       1        Lat_Site, Lon_Site
        real*8    Lalt,t,h,x,f,err,pitch,roll,p0,mu,R,M,S
        real*8  Omega, MAC, Lat_Site, Lon_Site
        real*8  Lon_Plus,Lon_Minus,Lat_Plus,Lat_Minus
        integer nxt,n,count
        open(unit= 8,file='c:\afit\contrl.out',status='unknown')
        open(unit= 11,file='c:\afit\AltVel.out',status='unknown')


c    constants: (in SI Units)
        mu = 3.98601d14
        R =  6378.135d3
        p0 = 101325.0d0
        M =  108864.0d0
        S =  249.90d0
        Omega = 0.000072921158560d0
        MAC = 12.0580d0
c        Minimum Altitude over center of runway for safe landing
c        estimated at 20,000 ft or 6098 meters
        Lalt = 6098
        count=0


c        open state vector produced by gravity turn program
        open(unit=10,file ='c:\afit\grav.out',status='unknown')

        do while(count.le.180)
        count = count + 1
c        read in state from gravity turn program to be used as initial state
c        (radius,longitude, latitude,velocity,flight path angle, heading angle)
        read(10,*) x(1,1),x(2,1),x(3,1),x(4,1),x(5,1),
       1    x(6,1)
        x(2,1)= x(2,1)/57.29578
        x(3,1)= x(3,1)/57.29578


c        Input the longitude and latitude of the east coast airport chosen for landing
c
        Lon_Site = -60.0/57.29578
        Lat_Site = 46.17/57.29578
c
```

```fortran
c          Zone for tracking intercept of airport longitude and latitude by +/- .05
c
           Lon_Plus = Lon_Site + .05/57.29578
           Lon_Minus = Lon_Site - .05/57.29578
           Lat_Plus = Lat_Site + .05/57.29578
           Lat_Minus = Lat_Site - .05/57.29578

c          for 45 degree pitch and 45 degree roll
c
           pitch = 0.78540d0
           roll = 0.78540d0
           n = 6
           h = .05d0
           t = 0.d0
           nxt = 0
           call haming(nxt)
            if(nxt.eq.0)then
              write(*,*) 'dead'
              stop
             endif
           do while(x(1,nxt)-R.gt.Lalt)
            call haming(nxt)

             if(dabs(t-INT(t+.01)).lt..025)then
             write(*,*)    count
             write(8,*) x(2,nxt)*57.29578,x(3,nxt)*57.29578
             if(x(2,nxt).gt.Lon_Minus.and.x(2,nxt)
     1         .lt.Lon_Plus.or.x(3,nxt).gt.Lat_Minus.and.
     2         x(3,nxt).lt.Lat_Plus)then
             write(11,*)     x(2,nxt)*57.29578,x(3,nxt)*57.29578,x(1,nxt)-R
     1                       ,x(4,nxt),t,count
             if(x(1,nxt).le.Lalt)then
             write(11,*) x(1,nxt)-R, x(4,nxt),t,count
             if(x(2,nxt)*557.29578.gt.-65)then
             stop
             endif
             endif
             endif
             endif
           end do
           end do
           close(10)
           stop
           end
```

Subroutine Dynam(nxt)

```
c
c     3D reentry over a spherical rotating earth.  Vinh pp 21-27
c     calculates eom assuming controls
c     are angle of attack and roll
c
c     state is radius r, longitude, latitude, speed, flight path
c     angle, and heading angle
c

         implicit Real*8 (a-h,o-z)

         common /dyn/g,icl,clnew,clmax,rho,cdmax,
     1         aoamax,icount
          common /param/ Lalt,pitch,roll,p0,mu,R,M,S,Omega,MAC,
     1          Lat_Site, Lon_Site

         common /ham/ t,x(230,4),f(230,4),err(230),n,h,mode
         double precision dynpre
         real*8 t,x,f,err,h,g
         double precision cd,cl
         integer n,nxt
         double precision clmax,clnew,cdmax,aoamax
         double precision Lon_Site, Lat_Site, l, Psi
         double precision Lalt,pitch,roll,p0,mu,R,M,S,Omega, MAC
         double precision rho,alt

c     constants
            mu  is the Gravitational constant of the Earth
c            R is Radius of the Earth
c            P0 is  pressure at sea level
c            M is mass of vehicle
c            S is vehicle reference area
c            Omega is rotation rate of the Earth

c     inverse square gravity
      g = mu/( x(1,nxt)*x(1,nxt) )

c
c      determine controls at this state

c     call contrl(nxt,cd,cl,dcddp,dcldp,rho,drhodr)
            alt = x(1,nxt)-R
c
        call atm(alt,p0,PALT,TALT,rho,dDdr,d2Ddr,sonic,mfp,dmfpdr)
            Kn = mfp/MAC

        call aero(pitch,Kn,cd,cl,cdp,clp)

c      dynamic pressure = rho S v^2 / 2
c
        dynpre = rho * S * x(4,nxt)*x(4,nxt) / 2.d0
```

```
c   equations of motion
c
C     f(1,nxt) - D(R)/DT - radius
C     f(2,nxt) - D(theta)/DT - longitude
C     f(3,nxt) - D(phi)/DT - latitude
C     f(4,nxt) - D(vel)/DT - velocity
C     f(5,nxt) - D(gamma)/DT - flight path angle
C     f(6,nxt) - D(psi)/DT - heading angle


c     This portion is only used in the Abort2 Program
c          This is to control the Orbiter to fly straight when it is at the
c     proper heading angle to intercept the selected airport
c
          l = dacos(dcos(1.5708-x(3,nxt))*dcos(1.5708-Lat_Site)
     1     + dsin(1.5708-x(3,nxt))*dsin(1.5708-Lat_Site)
     2     * dcos(x(2,nxt)-Lon_Site))

          Psi = dasin((dsin(1.5708-x(3,nxt))*dsin(x(2,nxt)-Lon_Site))
     1     /dsin(l))

          if(x(6,nxt).ge.(Psi+1.5708))then
          roll = 0.0d0
          pitch = 0.5236d0
          endif

c   Vinh eqns 2-28

          f(1,nxt) = x(4,nxt)*dsin(x(5,nxt))

          f(2,nxt) = x(4,nxt)*dcos(x(5,nxt))*dcos(x(6,nxt))/(x(1,nxt)
     1     *dcos(x(3,nxt)))

          f(3,nxt) = x(4,nxt)*dcos(x(5,nxt))*dsin(x(6,nxt))/x(1,nxt)
c
c   Vinh eqns 2-31; all coriolis terms retained

          f(4,nxt) = - CD*dynpre/M - g*dsin(x(5,nxt))

          f(5,nxt) = CL*dynpre*dcos(roll)/(M*x(4,nxt))
     1     -(G - (x(4,nxt)*x(4,nxt)/x(1,nxt)) )
     2     *dcos(x(5,nxt))/x(4,nxt)
     3     + 2.d0*Omega*dcos(x(3,nxt))*dcos(x(6,nxt))

          f(6,nxt) = -x(4,nxt)*dcos(x(5,nxt))*
     1     dcos(x(6,nxt))*dtan(x(3,nxt))/x(1,nxt)
     2     + CL*dynpre*dsin(roll)/(M*dcos(x(5,nxt))*x(4,nxt))
     3     + 2.d0*Omega*( dtan(x(5,nxt))*dcos(x(3,nxt))*dsin(x(6,nxt))
     4     - dsin(x(3,nxt)))
c     if(mode .eq. 0) return


          return
          end
```

Subroutine Aero(alpha, Kn, Cd, Cl, Cdp, Clp)

```
c
c     aerodynamic model for the shuttle, Blanchard et al, JSR 31, 550
c     converted to angle of attack alpha in radians; outputs Cd, Cl not
c     Ca, Cn; also returns first derivatives.
c
      double precision alpha,Kn,Cd, Cl, Cdp, Clp
      double precision Cnc, Cac, Cnf, Caf, Cncp, Cacp, Cnfp, Cafp
      double precision Cnbar, Cabar,alpha2,alpha3
      double precision Ca,Cn,Cap,Cnp,Cdf,Clf
c
      alpha2 = alpha * alpha
      alpha3 = alpha2 * alpha
c
c     hypersonic continuum
c
      Cnc = -0.839782d0 + 3.0012d0 * alpha - 0.303891d0 * alpha2
      Cac = -0.0086314d0 + 0.190247d0 * alpha - 0.220613d0 * alpha2
     1     + 0.110351d0 * alpha3
c
      Cncp = 3.0012d0 - 0.607781d0*alpha
      Cacp = 0.190247d0 - 0.441227d0 * alpha + 0.331053d0 * alpha2
c
c     free molecular flow
c
      Cnf = 0.00158739d0 + 0.526217d0 * alpha + 3.17184d0 * alpha2
     1     - 1.34772d0 * alpha3
      Caf = 0.751105d0 + 0.944601d0 * alpha + 1.94409d0 * alpha2
     1     - 2.20399d0 * alpha3
c
      Cnfp = 0.526217d0 + 6.34367d0 * alpha - 4.04317d0 * alpha2
      Cafp = 0.944601d0 + 3.88819d0 * alpha - 6.61198d0 * alpha2
c
c     bridging coefficients
c
      if( dlog10( Kn ) .lt. 1.3849d0 ) then
         Cnbar = dexp( -0.29981d0 * ( 1.3849d0 - dlog10( Kn ) )**
     1           1.7128d0 )
      else
         Cnbar = 1.d0
      endif
c
      if( dlog10( Kn ) .lt. 1.2042d0 ) then
         Cabar = dexp( -0.2262d0 * ( 1.2042d0 - dlog10( Kn ) )**
     1           1.8410d0 )
      else
         Cabar = 1.d0
      endif
```

```
c     merged normal and axial coefficients
c
      Cn = Cnc + ( Cnf - Cnc )*Cnbar
      Ca = Cac + ( Caf - Cac )*Cabar
c
      Cnp = Cncp + ( Cnfp - Cncp ) * Cnbar
      Cap = Cacp + ( Cafp - Cacp ) * Cabar
c
c     convert to Cl, Cd
c
      Cd =  Ca * dcos(alpha) + Cn * dsin(alpha)
      Cl = -Ca * dsin(alpha) + Cn * dcos(alpha)
c
      Cdp = Cap * dcos(alpha) + Cnp * dsin(alpha)
     1     -Ca * dsin(alpha) +  Cn * dcos(alpha)
      Clp = -Cap* dsin(alpha) + Cnp * dcos(alpha)
     1     -Ca * dcos(alpha) - Cn  * dsin(alpha)
c
      return
      end
```

Bibliography

1. Anderson, John D., Jr . *Fundamentals of Aerodynamics.* . New York: The McGraw-Hill Companies, 1991.

2. Bate, Roger R., Mueller, Donald D., and White, Jerry E. *Fundamentals of Astrodynamics.* New York: Dover Publications, 1971

3. Blanchard, Robert C., Kevin T. Larman, and Christina D. Moats. "Rarefied-Flow Shuttle Aerodynamics Flight Model", *Journal of Spacecraft and Rockets, 31*: 550-556 (August 1994).

4. Department of Defense. *High Altitude Canada.* DOD Flight Information Publication. St. Louis: National Imagery and Mapping Agency, 14 August 1997.

5. Department of Defense. *High Altitude United States.* DOD Flight Information Publication. St. Louis: National Imagery and Mapping Agency, 14 August 1997.

6. Dumoulin, Jim., "STS News Reference Manual." WWWeb, http://www.ksc.nasa.gov/shuttle/technology/sts-newsref (11 Oct 95).

7. Hill, Philip G. and Peterson, Carl R. *Mechanics and Thermodynamics of Propulsion.* Reading MA: Addison-Wesley Publishing Company, 1992.

8. Pranke, Dirk E. and Wertz, James R. *Compendium of Spherical Trigonometry Formulas.* Torrance: Space Mission Engineering, Microcosm Inc., October 1991.

9. Regan, Frank J. and Anandakrishnan, Satya M. *Dynamics of Atmospheric Reentry.* Washington DC: American Institute of Aeronautics and Astronautics, 1993.

10. *Space Shuttle.* National Aeronautics and Space Administration. U.S. Government Printing Office: 1975.

11. Vinh, Nguyen X., Busemann, Adolf, and Culp, Robert D. *Hypersonic and Planetary Entry Flight Mechanics.* Ann Arbor: The University of Michigan Press, 1980.

12. Wiesel, William E., Jr. Class handout distributed in MECH 636, Advanced Astrodynamics. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, January 1997.

13. Wiesel, William E., Jr. *Spaceflight Dynamics.* New York: The McGraw-Hill Companies., 1997.

Vita

Capt Rich Neufang was born on 11 November 1969 in Waynesboro, Pennsylvania. He graduated from high school in 1988 from Conner High School in Hebron Kentucky . In the fall of the same year he attended undergraduate studies at Embry-Riddle Aeronautical University in Daytona Beach, Florida. He graduated with a Bachelor of Science degree in Aerospace Engineering and received his commission as a second lieutenant in the United States Air Force in April 1993.

His first assignment was at Wright Patterson AFB as an Acquisition Officer for the C-17 System Program Office. During this tour of duty he worked on various systems including Cargo Compartment Heat and the On-Board Inert Gas Generating System (OBIGGS). In May 1996, he entered the School of Engineering, Air Force Institute of Technology.

Permanent Address: 1846 Timber Lane
Burlington, KY 41005

REPORT DOCUMENTATION PAGE

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE December 1997	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE

SPACE SHUTTLE EAST COAST ABORT MODES
FOR HIGH INCLINATION LAUNCHES

5. FUNDING NUMBERS

6. AUTHOR(S)

Richard K. Neufang, Capt, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
2750 P Street
WPAFB OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GA/ENY/97D-3

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Lt. Col. Scott J. Horowitz
CB
2101 NASA Road One
Houston, TX 77058

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

A

13. ABSTRACT *(Maximum 200 words)*

This study investigated the possibility of abort landing the Space Shuttle at east coast airports when launched at inclinations of 51.6 degrees or more. Computer modeling was used to characterize both the Shuttle launch out of Cape Canaveral and two methods of un-powered abort descents from various points in the launch following Solid Rocket Booster (SRB) separation. The first method used varying values of pitch and roll held constant to control the descent. By plotting the latitude and longitude of the point in the descent when the nominal landing altitude was achieved against locations of east coast airports it was found that there are indeed east coast abort opportunities for high inclination launches out of Cape Canaveral. The second method used a constant pitch and roll until the proper heading angle to intercept a desired target airport was reached then maneuver to 0 degrees roll and 30 degrees pitch. These trajectories were attempted throughout the launch for different airports so that windows of opportunity could be established. It was shown that these windows exist but only for limited times ranging from 8 to 34 seconds. These opportunities may be expanded if further studies investigate powered and optimal control cases.

14. SUBJECT TERMS

Gravity-Turn Trajectory, Abort Modes, Hypersonic Re-
entry, Rarefied Flow Shuttle Aerodynamics Model

15. NUMBER OF PAGES
90

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL